Emulex Drivers for Solaris

FC and FCoE version 2.50o

NIC version 1.10c

User Manual

P004439-01A Rev. A

One Network.
One Company.
Connect with Emulex.

Information furnished by Emulex is believed to be accurate and reliable. However, no responsibility is assumed by Emulex for its use; or for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright or related rights of Emulex.

Emulex provides this manual "as is" without any warranty of any kind, either expressed or implied, including but not limited to the implied warranties of merchantability or fitness for a particular purpose. Emulex may make improvements and changes to the product described in this manual at any time and without any notice. Emulex assumes no responsibility for its use, nor for any infringements of patents or other rights of third parties that may result. Periodic changes are made to information contained herein; although these changes will be incorporated into new editions of this manual, Emulex disclaims any undertaking to give notice of such changes.

Emulex, 3333 Susan Street
Costa Mesa, CA 92626

Introduction 1
Important Considerations 1
New in This Release 1
Known Issues 1
Driver Information (emlxs and oce) 1
Installing the Driver 2
Downloading and Installing the Driver for Solaris 10 and OpenSolaris (SPARC, x64 and x86) 2
Uninstalling the Driver 2
Utilities 3
The HBAnyware Utility 3
The emlxadm Utility 3
The emlxdrv Utility 3
Installing the FCA Utilities and the HBAnyware Utility 4
Installing or Updating the FCA Utilities Using the emlxu_install Script 4
Installing the HBAnyware Utility, Web Launch and Security Configurator 6
Installing or Updating the Utilities Package Manually 8
Removing the Utilities Using the emlxu_remove Script 9
Removing the Utilities Package Manually 10
Configuration and Migration 11
Introduction 11
Changing Driver Properties Using the emlxs.conf File 11
Enabling NPIV Support on Solaris 10 11
Enabling NPIV Support on OpenSolaris 12
NPIV Configuration Limits 12
NPIV and OS Virtualization 12
Using VPorts with Logical Domains, Containers and xVM 13
Configuring Target Mode Support for OpenSolaris 13
Emulex SFS FCA Driver Properties 13
The Configuration File (emlxs.conf) 13
Configuring the NIC Driver 18
Configuring the NIC Interface 19
dladm Support on OpenSolaris 20
Migrating from the Solaris LPFC Driver to the Solaris emlxs Driver 20
Operational Behaviors of the emlxs Driver 20
Sample Script File Details 21
start_emlxs_migration.sh 21
finish_emlxs_migration.sh 22
Migrating a Configuration without FC Boot 23
Migrating Automatically 23
Prerequisites 23
Procedures 23
Migrating a Configuration with FC Boot 25
Prerequisite 25
Procedure 25
Solaris emlxs and Solaris LPFC Driver Properties Cross-Reference Table 28
Console and Log Messages 34
emlxs Logs 34
General Situations 34

Messages 34
Severity Levels 36
Message Log Example 36
Miscellaneous Events 37
Driver Events 38
HBA Initialization Events 40
Memory Management Events 41
Service Level Interface (SLI) Events 42
Mailbox Events 46
Node Events 47
Link Events 48
ELS Events 49
General I/O Packet Events 51
FCP Traffic Events 53
FCT Traffic Events 53
IP Traffic Events 54
Solaris SFS Events 55
IOCTL Events 56
Firmware Download Events 57
Common Transport Events 59
FCSP (Fibre Channel Security Protocol) Events 60

oce Logs 62

Introduction

This document provides the information needed to use the Emulex FC driver for Solaris StorEdge SAN Foundation Software (SFS), which is part of the SFS (Leadville) stack. The module name for the Emulex FC driver for Solaris SFS is ‘emlxs’. This document also provides information about the Emulex NIC driver for Solaris which is used to operate the NIC function of the Emulex OneConnect™ Universal Converged Network Adapter (UCNA). The module name for the NIC driver is 'oce'.

To work with the drivers for Solaris the system administrators should be familiar with Solaris and have access to standard system information. For the FC driver, familiarity with Sun StorageTek SFS and nature and use of Fibre Channel (FC) is essential. For the NIC driver, familiarity with the Emulex OneConnect UCNA and Ethernet networking is essential.

Important Considerations

New in This Release

- Supports OneConnect adapters.
- Supports the Ethernet NIC protocol.

Known Issues

See the product release notes for the latest information.

Driver Information (emlxs and oce)

Prerequisites

One of the following operating systems must be installed:

- Solaris 10 SPARC
- Solaris 10 x64 and x86
- OpenSolaris for SPARC
- OpenSolaris for x64 and x86

Compatibility

For a list of adapters that are compatible with the emlxs and oce driver, see the driver's Downloads page on the Emulex Web site. For compatible firmware versions, see the Downloads page for the specific adapter.

Note: Refer all LP21000 support issues to Emulex, not Sun Microsystems.

Installing the Driver

Downloading and Installing the Driver for Solaris 10 and OpenSolaris (SPARC, x64 and x86)

The Solaris SFS FCA (emlxs) driver and the Solaris OneConnect UCNA (oce) driver are distributed by Sun as part of the Solaris 10 operating environment, with driver updates distributed as part of Solaris updates and patches. If the Solaris SFS FCA (emlxs) driver and the Solaris OneConnect UCNA (oce) driver are not already installed, obtain and install the Solaris 10 or OpenSolaris packages.

To obtain and install the Solaris packages:

1. Go to http://www.sun.com/download/products.xml?id=42c4317d and click **Download**.
2. Log in with your user name and password, and accept the license agreement.
3. Select and download the driver package.
4. Select and download the readme file, and follow its instructions.

To finish the installation (or if the Solaris SFS FCA driver/ the Solaris OneConnect UCNA NIC driver was already installed), install the driver by obtaining and installing individual patches:

1. Go to http://sunsolve.sun.com/pub-cgi/show.pl?target=patchpage. Enter and download the following required patches:
   - For Solaris 10 SPARC systems:
     - 141876
   - For Solaris 10 x64 and x86 systems:
     - 141877
   - For OpenSolaris
     - Visit www.opensolaris.org for the latest patches.
2. Follow the instructions to install each patch.

Uninstalling the Driver

To uninstall the driver:

1. Remove the driver patch by typing:

   ```
   patchrm <patch_id>
   ```

   For example:

   ```
   # patchrm 139609-04
   ```

The script performs the removal and displays the following messages:

```
Validating patches...
Loading patches installed on the system...
Done!
Checking patches that you specified for removal.
Done!
Approved patches will be removed in this order:
139609-04
Checking installed patches...
Backing out patch 139609-04...
Patch 139609-04 has been backed out.
```

For additional information on installing and removing patches, see the Solaris system administration documentation and the patchadd(1M) and patchrm(1M) manual pages.

Utilities

Emulex provides three utilities to facilitate the configuration and use of the Solaris drivers.

- The HBAnyware utility (supports LightPulse HBAs but not OneConnect CNAs)
- The emlxadm utility
- The emlxdrv utility

The HBAnyware Utility

The HBAnyware utility provides all functions as emlxadm, and a number of additional ones, on multiple systems. It offers a choice of a graphical user interface and a scriptable command-line interface. It is intended to be a direct user interface to the Fibre Channel input/output (FCIO) interface provided by the Sun StorEdge SFS. The FCIO interface provides a Sun common ioctl interface to the FCTL, which manages the FCA drivers for each Fibre Channel adapter attached to the host system.

Note: The HBAnyware utility is supported only for Lightpulse adapters and not for OneConnect adapters. The Solaris FCA utilities are supported for all adapters.

The emlxadm Utility

The emlxadm utility is used to change driver parameters through a local interactive or command-line interface. It can also be used for firmware updates on non-Sun branded devices.

Note: emlxadm only updates the Fibre Channel ASIC firmware on LP21000/21002, not the IP/Ethernet.

The emlxdrv Utility

The emlxdrv utility temporarily associates the Emulex emlxs Solaris SFS driver and the Solaris LPFC driver to the various models of Emulex Fibre Channel adapters, for use during migration from the Solaris LPFC driver to the Solaris SFS driver. The emlxdrv utility is intended to be used for binding (associating) the Emulex emlxs (Leadville Fibre Channel) driver and the Emulex LPFC (traditional non-Leadville Fibre Channel) driver to the various models of Emulex Fibre Channel adapters. If the driver binding configuration is changed, the host system must usually be rebooted in order for the new configuration to take effect.

Installing the FCA Utilities and the HBAnyware Utility

Note: The HBAnyware utility is supported only for Lightpulse adapters and not for OneConnect adapters. The Solaris FCA utilities are supported for all adapters.

The FCA utilities and the HBAnyware configuration utility are packaged together in one application kit tar file. The FCA utilities is comprised of the emlxadm utility and the emlxdrv utility.

- The emlxadm utility provides an interface to the Fibre Channel input/output (FCIO) interface provided by the Sun StorageTek SFS.
- The emlxdrv utility temporarily associates or binds the Emulex emlxs Solaris SFS driver and the Solaris LPFC driver to the various models of Emulex FC adapters during migration from the Solaris LPFC driver to the Solaris SFS driver.

Refer to the *Solaris FCA Utilities User Manual* and the *HBAnyware version 4.1 User Manual* to learn how to use these utilities.

To install the utilities:

1. Log in as or su to ’root’.
2. Extract the FCA utilities and the HBAnyware utility files from the tar by typing:

   ```
   tar xvf Solaris-4.1a37-1.03h-1a.tar
   ```

   Both of the following tar kits will be placed in the specified directory:

   - Solaris-4.1a37-1.03h-1a.tar
   - Solaris-4.1a37-1.03h-1a-sparc.tar

   Each of these .tar files contains the following:

   - readme.first.txt
   - emlxu_kit-<version>-<platform>.tar - The emlxu_kit files contain the FCA utilities.
   - EmlxApps<version>-<platform>.tar - This file contains the HBAnyware utility.
3. Install the utilities. Type:

   ```
   ./install
   ```

   This installs the HBAnyware utility and emlxu_kit. HBAnywareSSC is not installed.

Installing or Updating the FCA Utilities Using the emlxu_install Script

Although it is possible to install emlxu onto one or more clients from a server, that procedure is not covered in this document; refer to the Solaris documentation.

Note: If an earlier version of the emlxu utilities package is already installed on the system, the emlxu_install script will remove the old version before installing the new one.

Prerequisites

Before installing the Emulex emlxu utilities package, you must completely install:

- The Emulex-Sun driver kit (SUNWemlxs) for Fibre Channel and Converged Network Adapters.

Procedure

To install the utilities kit using the emlxu_install script:

1. Untar the emlxu_kit-1.04i-<platform>.tar file.

   ```
   tar xvf emlxu_kit-1.04i-<platform>.tar
   ```

   The emlxu_install script is available.
2. Install the FCA utilities by typing:

   ```
   emlxu_install
   ```

   The script removes any earlier version of the emlxu utilities package. (If an earlier package is not found, this fact is indicated; skip to step 6.) The following text is displayed:

   ```
   <Removing old EMLXemlxu package>
   ```
3. If an old package is installed, you are prompted to remove it:

   ```
   Do you want to remove this package? [y,n,?,q]
   ```
4. Enter **<y>**. The following message is displayed:

   ```
   Removal of <EMLXemlxu> was successful.
   ```

   The script expands the utilities kit .tar file and begins installing the new package. A message similar to the following message is displayed:

   ```
   <Expanding emlxu_kit-1.04i-sparc.tar>
   <Adding new package>
   ```

   The script installs the emlxu utilities package. The package is prepared for installation and you are prompted for confirmation by the following message:

   ```
   Do you want to continue with the installation of <EMLXemlxu> [y,n,?]:
   ```
5. Enter **<y>**. The installation progress is indicated.
6. Examine the output for any errors or warnings. If the installation is successful, the following message is displayed near the end of the process:

   ```
   Installation of <EMLXemlxu> was successful.
   ```

   The script performs some cleanup and displays the following messages:

   ```
   <Cleaning directory>
   <emlxu_install complete>
   <Execute "emlxu_remove" when ready to uninstall>
   ```

   The script leaves a copy of the emlxu_remove script in the working directory with the original utilities kit tar file. You can remove this script, or leave it in the directory and use it to uninstall the emlxu utilities from your system in the future. See *Installing or Updating the Utilities Package Manually* on page 8 for more details.

The emlxu utilities installation is complete. The utility programs are located in the /opt/EMLXemlxu/bin directory.

You do not have to reboot the system to run a utility program, but you must either enter the program’s full path name, or add the package's bin directory (/opt/EMLXemlxu/bin) to the system environment’s search path. To use the man pages provided by the package, you must also add the package's man directory (opt/EMLXemlxu/man) to the system environment's man path.

For further information on installing and removing packages, refer the Solaris system administration documentation and the pkgadd(1M) and pkgrm(1M) manual pages.

Installing the HBAnyware Utility, Web Launch and Security Configurator

Note: The HBAnyware utility is supported only for Lightpulse adapters and not for OneConnect adapters. The Solaris FCA utilities are supported for all adapters.

Installing the HBAnyware Utility

Note: The HBAnyware utility is supported only for Lightpulse adapters and not for OneConnect adapters. The Solaris FCA utilities are supported for all adapters.

Prerequisites

- The FCA utilities must be installed prior to installing the HBAnyware utility.
- Java Runtime Environment:

  Version 5 of the Java Runtime Environment (JRE) must be installed. The HBAnyware utility will not run under earlier versions of the JRE.

Caution: The utilities require the java runtime binaries and libraries, so their path must be included at the beginning of the PATH environment variable to avoid conflicts with possible earlier versions of java that may still be installed on the system. For example, if the java runtime binaries are in /usr/java/bin, then include this path in the PATH environment variable.

For example: (bash> export PATH="/usr/java/bin:$PATH")

The JRE and instructions for installation can be found at http://java.sun.com/downloads/index.html.

Procedure

To install the HBAnyware utility from the tar file:

1. Untar the EmlxApps tar file, type:

   ```
   tar xvf EmlxApps<version>-<platform>.tar
   ```

2. Unzip the HBAnyware package file. Type:

   ```
   gunzip HBAnyware-<version>-<platform>.tar.gz
   ```

3. Untar the HBAnyware package file. Type:

   ```
   tar -xvf HBAnyware-<version>-<platform>.tar
   ```

4. Run the pkgadd utility. Type:

   ```
   pkgadd -d
   ```

5. When prompted by pkgadd, choose to install the HBAnyware utility and proceed by answering the installation questions.

Installing the HBAnyware Utility with Web Launch

Note: The HBAnyware utility is supported only for Lightpulse adapters and not for OneConnect adapters. The Solaris FCA utilities are supported for all adapters.

Prerequisites

Before installing the HBAnyware configuration utility with Web Launch, ensure your systems meet the following requirements.

Note: This information is also available online in the /opt/HBAnyware/README_WEBLAUNCH.txt file.

- The system on which you are installing the Web Launch services package (the server) requires that the HTTP Web server be configured to handle the JNLP MIME file type. Follow these steps:
  - a Change the working directory to the directory containing the Apache configuration files for example: /etc/apache or /etc/apache2).
  - b Edit the file “mime.types”.
  - c Add the following line to the file:

    ```
    application/x-java-jnlp-file jnlp JNLP
    ```
  - d Save the file.
  - e Stop and restart the HTTP Web server (to enable the Web server to detect this change).
- The system on which you are running the browser (the client) requires the Java Runtime Environment (JRE) 5.0 or later be installed. Below are the specific requirements:
  - Sun 32-bit JRE 5.0 or later for Intel based systems (x86 and IA64)
  - Sun 32-bit JRE 5.0 or later for x86-64

  Refer to the appropriate vendor documentation for detailed instructions about configuring and starting the HTTP server and installing the JRE.
- The HBAnyware utility must be installed before installing HBAnyware with Web Launch.

Procedure

To install HBAnyware with Web Launch:

1. Log in as ‘root’.
2. Navigate to the HBAnyware directory. Type:

   ```
   cd  /opt/HBAnyware
   ```
3. Run the install script. Type:

   ```
   ./wsinstall
   ```
4. When prompted, enter the Web server's document root directory. For example:

   ```
   /usr/apache/htdocs
   ```
5. You are provided with the IP address of the host and asked if that is the IP address that is being used by your Web server. Answer <y> or <n> as appropriate. If you answer <n>, you are prompted for the IP address you want to use.
6. You are asked if your Web server listening on the normal default HTTP port (80)? Answer <y> or <n> as appropriate. If you answer <n>, you are prompted for the port you wish to use.

   You are notified that the installation of the HBAnyware Web Launch package has completed.

Installing the HBAnyware Utility Security Configurator

Note: The HBAnyware utility is supported only for Lightpulse adapters and not for OneConnect adapters. The Solaris FCA utilities are supported for all adapters.

Follow these instructions to install the Security Configurator on your system.

Prerequisites

- The HBAnyware utility must be installed on the system.
- Java Runtime Environment:

  Version 5 of the Java Runtime Environment (JRE) must be installed. The HBAnyware configuration utility will not run under earlier versions of the JRE.

  The JRE and instructions for installation can be found at:
  http://java.sun.com/downloads/index.html.

Procedure

To install the HBAnyware utility Security Configurator from a tar file:

1. Untar the EmlxApps tar file, type:

   ```
   tar xvf EmlxApps<version>-<platform>.tar
   ```

2. Unzip the HBAnywareSSC package file:

   ```
   gunzip HBAnywareSSC-<version>-<platform>.tar.gz
   ```

3. Untar the HBAnywareSSC package file. Type:

   ```
   tar xvf HBAnywareSSC-<version>-<platform>.tar
   ```

4. Run the pkgadd utility.

   ```
   pkgadd -d
   ```

5. When prompted by pkgadd, choose to install HBAnywareSSC and answer the HBAnywareSSC installation questions.

Installing or Updating the Utilities Package Manually

Compatibility

- Emulex SFS drivers have been supporting corresponding versions of the HBAnyware utility starting with Solaris 10 update 4 (driver version 2.20).
- The current version of the HBAnyware utility (HBAnyware 4.1) is compatible with both the Solaris 10 Update 8 driver (2.40) and the previous Solaris 10 Update 7 driver (2.31).

Prerequisites

- If an earlier version of the emlxu utilities package is already installed on the system and you want to install a different version, follow the instructions in *Removing the Utilities Package Manually* on page 10, then return to this section to install the new utilities package.

Procedure

To install the emlxu utilities package manually:

1. Log in as or su to ’root’.
2. Copy the utilities kit from the distribution medium into a directory, referred to here as <directory>. The utilities kit is a .tar file named something similar to emlxu_kit-1.01c-sparc.tar.

3. Change to the directory where you put the kit tar file by typing:

   ```
   cd <directory>
   ```

4. Extract the installation images from the tar file by typing:

   ```
   tar xvf emlxu_kit-1.04i-sparc.tar
   ```

5. Install the EMLXemlxu utilities package by typing:

   ```
   pkgadd -d . EMLXemlxu
   ```

   The package is prepared for installation, and you are prompted to confirm the installation with the following message:

   ```
   Do you want to continue with the installation of <EMLXemlxu> [y,n,?]
   ```

6. Enter **<y>**. The installation progress is indicated.
7. Examine the output for any errors or warnings. If the installation is successful, the following message is displayed near the end of the process:

   ```
   Installation of <EMLXemlxu> was successful.
   ```

The emlxu utilities installation is complete. The utility package's programs are located in the /opt/EMLXemlxu/bin directory.

You do not have to reboot the system to run a utility program, but you must either enter the program’s full path name, or add the package's bin directory (/opt/EMLXemlxu/bin) to the system environment’s search path. To use the man pages provided by the package, you must also add the package's man directory (opt/EMLXemlxu/man) to the system environment's man path.

Removing the Utilities Using the emlxu_remove Script

You can uninstall the utilities kit using the emlxu_remove script. If you do not have the emlxu_remove script and you do not have the original emlxu utilities kit tar file, you must uninstall the emlxu package manually; follow the instructions in *Removing the Utilities Package Manually* on page 10. If you are updating the emlxu utilities to a newer version and you have the new utilities kit tar file, you do not need to use the emlxu_remove script; the emlxu_install script removes any old version as it installs the newer version; see *Installing or Updating the FCA Utilities Using the emlxu_install Script* on page 4 for more details.

To uninstall the utilities package (without updating them):

Note: All emlxu files are removed.

1. Go to the directory where the emlxu_remove script is located, or to the directory where the original utilities kit tar file is located, by typing:

   ```
   cd <directory>
   ```

2. If you have the emlxu_remove script, skip to step 3. If you do not have the emlxu_remove script but you do have the original emlxu utilities kit tar file, extract the emlxu_remove script from the tar file by typing:

   ```
   tar xf emlxu_kit-1.04i-sparc.tar emlxu_remove
   ```

3. Remove the emlxu utilities package by typing:

   ```
   emlxu_remove
   ```

   The script locates the EMLXemlxu utilities package, and the following message is displayed:

   ```
   <Removing EMLXemlxu package>
   ```

Note: If no package is installed, the following message is displayed:

```
pkgrm: ERROR: no package associated with <EMLXemlxu>
```

You are prompted to remove the package with the following message:

```
Do you want to remove this package? [y,n,?,q]
```

4. Enter **<y>**. The following message is displayed:

```
Removal of <EMLXemlxu> was successful.
```

The script performs some cleanup and displays the following message:

```
<Removing emlxu scripts>
<emlxu_remove complete>
```

The utilities package is removed. If you want to install another version of the emlxu utilities package, do so now by following the instructions in one of the following sections:

- *Installing or Updating the FCA Utilities Using the emlxu_install Script* on page 4
- *Installing or Updating the Utilities Package Manually* on page 8

For additional information on installing and removing packages, refer the Solaris system administration documentation and the pkgadd(1M) and pkgrm(1M) manual pages.

Removing the Utilities Package Manually

To remove the emlxu utilities package:

1. Remove the EMLXemlxu utilities package by typing:

```
pkgrm EMLXemlxu
```

You are prompted to confirm the removal by the following message:

```
Do you want to remove this package? [y,n,?,q]
```

2. Enter **<y>**. The package is prepared for removal, and you are prompted again for confirmation:

```
Do you want to remove this package? [y,n,?,q]
```

3. Enter **<y>**. The following message is displayed:

```
Removal of <EMLXemlxu> was successful.
```

For additional information on installing and removing packages, refer the Solaris system administration documentation and the pkgadd(1M) and pkgrm(1M) manual pages.

Configuration and Migration

Introduction

‘emlxs’ is the module name for the Emulex SFS FCA driver. You can configure the Emulex SFS FCA driver properties by:

- Editing the emlxs.conf file which is described in this section.
- Using the HBAnyware configuration utility. Refer to the *HBAnyware 4.1 User Manual* for more information.
- Using the Emulex FCA utilities. Refer to the *Solaris FCA Utilities User Manual* for more information.

If the Emulex LPFC driver for Solaris is already installed, you can migrate to the emlxs driver either by customizing and running the unsupported sample scripts provided by Emulex, or by manually performing a set of procedures. Refer to *Configuring the NIC Driver* on page 18 for more information.

Changing Driver Properties Using the emlxs.conf File

The emlxs.conf file contains all the properties necessary to initialize the Emulex SFS FCA driver. The emlxs.conf driver properties are described in Table 1 on page 14.

In the emlxs.conf file, all adapter-specific properties have an emlxs*X*-prefix (where *X* is the driver instance number). For example, setting emlxs0-link-speed=4 makes 4 the default link speed setting for the zero instance of the driver. Changes to the emlxs.conf file require you to unload and reload the driver.

To change driver properties:

1. Open the emlxs.conf file in a text editor.
2. Change the properties you want.
3. Save the file.
4. See Table 1 on page 14 for activation requirements.

Enabling NPIV Support on Solaris 10

To enable NPIV support in the driver:

1. Login as or su to ’root’.
2. Set enable-npiv to 1 in the emlxs.conf file.

3. The fp driver properties are updated when the EMLXemlxu is installed. Entries from 2 to 255 are added to the `/kernel/drv/fp.conf` file. For example:

```
name="fp" class="fibre-channel" port=0;
name="fp" class="fibre-channel" port=1;
name="fp" class="fibre-channel" port=2;
name="fp" class="fibre-channel" port=3;
name="fp" class="fibre-channel" port=4;
name="fp" class="fibre-channel" port=5;
name="fp" class="fibre-channel" port=6;
name="fp" class="fibre-channel" port=7;
name="fp" class="fibre-channel" port=8;
name="fp" class="fibre-channel" port=9;
```

   The first two lines are default. By adding port 2 to 9 to support up to 10 virtual ports. The port number of each entry must be in order with no gaps in between.

4. Reboot the system.

To create, delete and list virtual ports after a system reboot, refer to the *HBAnyware 4.1 User Manual*.

Enabling NPIV Support on OpenSolaris

To enable NPIV support in the driver:

1. Login as or su to ’root’.
2. Set enable-npiv to 1 in the emlxs.conf file (This is set to 0 by default.)
3. Reboot the system. If enable-npiv is already set, do not reboot.
4. Refer to Chapter 6 of the "Solaris Express SAN Configuration and Multipathing Guide". The guide is available at http://docs.sun.com/app/docs/doc/820-3070?l=en&q=fcadm

NPIV Configuration Limits

The following limitations apply to NPIV:

- There is no FC-IP support on virtual ports.
- You cannot delete a virtual port with a mounted file system.
- Due to the limitation of Solaris Leadville stack, deleting a virtual port causes that virtual port to go offline.
- The Emulex Light Pulse LP11000 and LPe11000 family of adapters can support up to 100 virtual ports.
- The Emulex Light Pulse LPe12000 family of adapters can support up to 255 virtual ports.

NPIV and OS Virtualization

Currently Solaris has many different OS Virtualization solutions such as LDOM, xVM, containers, zones, and so on. Devices configured to be seen on an Emulex Fibre Channel or Ethernet port (either a physical port or a virtual port) can be used with any of these OS Virtualization solutions. Emulex strongly recommends that you consult the latest document on these technologies to learn the best use of resources related to NPIV technology.

Using VPorts with Logical Domains, Containers and xVM

Using NPIV with logical domains, Solaris containers and xVM user domains is simple.

1. Create virtual ports for the domains/containers to which you want to present dedicated storage.
2. Discover and attach the targets to virtual ports.
3. Assign the target to the domain or container. The attachment runs through the virtual port which provides the path to the target.

Configuring Target Mode Support for OpenSolaris

To configure target mode support for OpenSolaris:

1. Login as or su to ’root’.
2. Set target-mode to “1” in the emlxs.conf file. You can also set individual paths to Target Mode:

   emlxsX-target-mode=1;

   Where X is the specific numeric path. For example, when emlxs1 is set to target mode, all other paths will stay in initiator mode.
3. Reboot the system.

To configure targets, refer to the SUN COMSTAR Administration document at
http://wikis.sun.com/display/OpenSolarisInfo/comstar+Administration

Emulex SFS FCA Driver Properties

- The emlxs.conf file contains all the properties necessary to initialize the Emulex SFS FCA driver.
- The HBAnyware configuration utility reflects the emlxs.conf driver properties. See the *HBAnyware version 4.1 User Manual* for more information about using HBAnyware with the Emulex SFS FCA driver.
- All properties are adapter specific.

Note: If any of the default property values were changed, verify that this change will not impact the migration **before** you migrate.

The Configuration File (emlxs.conf)

In the emlxs.conf file, all adapter-specific parameters have emlxsX-prefix (where X is the driver instance number); for example, setting emlxs0-link-speed=4 makes 4 Gb/s the default link speed.

Changes to the emlxs.conf file require you to unload and reload the driver.

Note: If you want to override a driver parameter for a single driver-loading session, you can specify it as a parameter to the modload command. For example: # modload /kernel/drv/emlxs automap=0 (for 32-bit platforms) or modload /kernel/drv/sparcv9/emlxs automap=0 (for 64-bit platforms).

Table 1: emlxs.conf Parameters

Property Name	Default	Min	Max	Activation	Comments
ack0	0	0	1	Adapter reset	Use ACK0 for class 2. If ACK0 is 1, the adapter tries to use ACK0 when running Class 2 traffic to a device. If the device doesn't support ACK0, then the adapter uses ACK1. If ACK0 is 0, only ACK1 is used when running Class 2 traffic.
adisc-support	1	0	2	Dynamic	Sets the level of driver support for the FC ADISC login I/O recovery method. 1= Partial support. Flush I/O's for non-FCP2 target devices at link down 0 = No support. Flush active I/O's for all FCP target devices at link down. 2 = Full support. Hold active I/O's for all devices at link down.
assign-alpa	0x00	0x00	0xef	Link reset	This parameter is only valid if topology is set to loop. A 0x00 setting means no preference. If multiple adapter instances on the same host are on the same loop, set this value differently for each adapter.
console-notices	0x00000000	0x00000000	0xFFFFFFFF	Dynamic	Verbose mask for notice messages to the console.
console-warnings	0x00000000	0x00000000	0xFFFFFFFF	Dynamic	Verbose mask for warning messages to the console.
console-errors	0x00000000	0x00000000	0xFFFFFFFF	Dynamic	Verbose mask for error messages to the console.
cr-count	1	1	255	Link reset	This value specifies a count of I/O completions after which an interrupt response is generated. This feature is disabled if cr-delay is set to 0.

Table 1: emlxs.conf Parameters (Continued)

Property Name	Default	Min	Max	Activation	Comments
cr-delay	0	0	63	Link reset	This value specifies a count of milliseconds after which an interrupt response generated if cr-count has not been satisfied. This value is set to 0 to disable the Coalesce Response feature as default.
enable-npiv	0	0	1	Adapter reset	Enables NPIV support in the driver.
link-speed	Auto-Detect	Auto-Detect, 1 Gb/s, 2 Gb/s, 4 Gb/s, 8 Gb/s		Link reset	Sets link speed for initializing FC connection.
linkup-delay	10	0	60	Adapter reset	Sets the linkup delay period (seconds) after adapter initialization.
log-notices	0xFFFFFFFF	0x00000000	0xFFFFFFFF	Dynamic	Verbose mask for notice messages to the messages file.
log-warnings	0xFFFFFFFF	0x00000000	0xFFFFFFFF	Dynamic	Verbose mask for warning messages to the messages file.
log-errors	0xFFFFFFFF	0x00000000	0xFFFFFFFF	Dynamic	Verbose mask for error messages to the messages file.
max-xfer-size	339968	131072	1388544	Reboot	Sets the maximum SCSI transfer size in bytes per IO. This parameter is only used by the driver on i386 platforms. The driver does not limit transfer size on SPARC platforms. This parameter determines the scatter gather list buffer size. A pool of buffers is reallocated by the driver during boot. A larger transfer size requires a larger memory allocation. Memory_model/max-xfer-size Small/131072 - 339968 Medium/339969 - 688128 Large/688129 - 1388544
network-on	0	0	1	Reboot	Enables/disables IP networking support in the driver.

Table 1: emlxs.conf Parameters (Continued)

Property Name	Default	Min	Max	Activation	Comments
num-iocbs	1024	128	10240	Adapter reset	This variable indicates the number of Input/Output Control Block (IOCB) buffers to allocate.
num-nodes	0	0	4096	Adapter reset	Number of FC nodes (NPorts) the driver-supports.
pci-max-read	2048	512	4096	Adapter reset	Sets the PCI-X max memory read byte count [512, 1024, 2048 or 4096]
pm-support	0	0	1	Reboot	Enable/Disable power management support in the driver. 0 = Disables power management support in the driver. 1 = Enables power management support in the driver.
ub-bufs	1000	40	16320	Reboot	Sets the number of unsolicited buffers to be allocated.
target-mode	0	0	1	Reboot	(OpenSolaris only) Enables/ disables COMSTAR target mode support. If target mode is enabled for that port, then SFS initiator mode is disabled for that port.
topology	0	0 =loop, then P2P 2 =P2P only 4 =loop only 6 =P2P, then loop		Link reset	Set to point-to-point mode if you want to run as an N_Port. Set to loop mode if you want to run as an NL_Port.

Table 1: emlxs.conf Parameters (Continued)

<table>
<tr><th>Property Name</th><th>Default</th><th>Min</th><th>Max</th><th>Activation</th><th>Comments</th></tr>
<tr><td>vport</td><td colspan="5">(Solaris 10 only) Virtual port registration table. The enable-npiv must be set to 1. The vport table may have any number of comma delimited entries. Each entry must be of the form:
"PHYS_WWPN:VPORT_WWNN:VPORT_WWPN:VPORT_ID"

PHYS_WWPN = World Wide Port Name of adapter's physical port
VPORT_WWNN = Desired World Wide Node Name of virtual port
VPORT_WWPN = Desired World Wide Port Name of virtual port
VPORT_ID = Desired virtual port ID (1 to max vports)
The port IDs must start at 1 and increment by 1 with no gaps in the count. The virtual port ID 0 is reserved for the physical port. Example:
vport= "10000000c9123456:28010000c9123456:20010000c9123456:1",
"10000000c9123456:28020000c9123456:20020000c9123456:2",
"10000000c9123457:28010000c9123457:20010000c9123457:1",
"10000000c9123457:28020000c9123457:20020000c9123457:2",
"10000000c9123457:28030000c9123457:20030000c9123457:3";
All entries are automatically created or removed by the HBAnyware Utility.</td></tr>
<tr><td>vport-restrict-login</td><td>1</td><td>0</td><td>1</td><td>Link reset</td><td>Sets the virtual port's behavior when discovering targets in the SAN. 1 prevents the VPort from logging into other initiator ports on the SAN. Also rejects logins from other ports in the SAN because it assumes that all ports that send a PLOGI are Initiators. When this parameter is turned off (0) the driver attempts to login to every port that it can access in the SAN and accept logins from all ports.
NOTE: In a SAN where there are other initiators this feature greatly reduces the hardware resources the driver uses.</td></tr>
</table>

Configuring the NIC Driver

The driver exports certain parameters that can be configured by editing the oce.conf file. If the oce.conf file is not present on the system, create the oce.conf file in the /kernel/drv/ directory.

To edit the oce.conf file:

1. Open the file in a text editor.
2. Make the relevant changes and save the file.

   The format of single lines in the file is as follows:

   <variable> = <value>;

   For example: oce_default_mtu = 9000;

   Comment lines must start with a '#' character.
3. If the driver is already loaded, unloaded the driver and re-load it for the changes to take effect. See “Uninstalling the Driver” on page 2 for more information.

The following parameters are configurable through the oce.conf file:

- oce_tx_ring_size - Number of descriptors in the transmit ring. It is set to 2048 (the maximum possible) by default but can be reduced to one of 256, 512 or 1024 if there are memory constraints on the system.
- oce_bcopy_limit - The largest sized packet that the driver will copy into descriptor buffers in the transmit path. This is an optimization feature and changing it will affect the performance of the UCNA. By default, it is set to 512 bytes.
- oce_default_mtu - The default MTU. The Emulex UCNA supports 1500 and 9000 byte MTU sizes. By default it is set to 1500.
- oce_lso_capable - Flag to enable/disable large send offload on the UCNA. By default it is set to 1 or ENABLE.
- oce_fm_capability - Sets the driver's fault management capability to one of the values defined for Solaris FM capability. It is a bitmap of one or more of the following values:

DDI_FM_NOT_CAPABLE	0x00000000
DDI_FM_EREPORT_CAPABLE	0x00000001
DDI_FM_ACCCHK_CAPABLE	0x00000002
DDI_FM_DMA_CHK_CAPABLE	0x00000004
DDI_FM_ERRCB_CAPABLE	0x00000008

  By default this value is set to 7. [See Solaris FMA documentation for more information].
- oce_log_level - Sets the driver's verbosity level in messages logged in /var/adm/messages. A higher verbosity level is intended for field logs and will affect hardware performance. By default, verbosity is set to 0 which is the least verbose.

  This value is of the form (MOD_MASK | SEVERITY) where

  MOD_MASK:

MOD_CONFIG	0x0001	Messages in the device configuration path are logged.
MOD_TX	0x0002	Messages in the transmit data path are logged.

MOD_RX	0x0004	Messages in the receive data path are logged.
MOD_ISR	0x0008	Messages in the interrupt path are logged.

SEVERITY:

CE_CONT	0	Continuation
CE_NOTE	1	Information
CE_WARN	2	Warning
CE_PANIC	3	Causes the OS to panic
CE_IGNORE	4	No action

The severity is one of the listed severity levels. For a set level of severity, only messages of that level and higher are logged. For example, if severity is set to CE_WARN, then messages with CE_CONT and CE_NOTE will not be logged.

The default value is severity of CE_WARN and MOD_MASK comprising of all modules.

Configuring the NIC Interface

Ensure that the NIC interface has been created using add_device.

To configure the NIC interface(s):

1. Plumb the interface.

   ```
   #ifconfig oce<X> plumb
   ```

   where ‘X’ is the interface number.

   To see the interfaces created, execute $> dladm show-link. This command lists all the interfaces in the system.
2. Assign an IP address.

   ```
   #ifconfig oce<X> <IP_Address> netmask <NetMask> up
   ```
3. Edit the /etc/hosts file (a symlink to /etc/inet/hosts) and add the IP address and hostname that you wish to assign to the given NIC interface. Refer hosts(4) for more information.
4. Edit the /etc/inet/ipnodes file and add an entry for the IP address and hostname for the given interface. The /etc/inet/ipnodes file is primarily for IPv6 only but this step is necessary for the IP address change to take effect.
5. Edit the /etc/netmasks file and add an entry with the IP address and desired subnet mask for the given interface. Refer netmasks(4) for more information.
6. Restart the network service. Use:

   ```
   $> svcadm restart network/physical
   ```

   or

   Reboot the system.

To remove the interface:

1. Perform the steps detailed in configuring the NIC interface in reverse order.
2. Remove all the entries in the files, and unplumb the interface, using the following command:

   ```
   #ifconfig oce<X> down unplumb
   ```

Alternatively, use sys-unconfig(1M) to delete the configurations of IP address, netmask, hostname, nfs mounts, ldap etc. on the host. The sys-unconfig utility will reboot the system and clear all existing IP configuration, so you will have to enter all the information again, even for the existing NICs already configured in the system. sys-unconfig must be executed from a console. On reboot, you are presented with a set of UI based data entry forms that facilitate the required change in configuration.

dladm Support on OpenSolaris

To configure the interface on OpenSolaris:

Use dladm to configure the interface and also perform runtime update of the following driver parameters:

- MTU - to change the MTU in OpenSolaris, unplumb the interface and execute

  ```
  $> dladm set-linkprop -p mtu=9000 <interface>
  ```

  where 'interface' is oce<0,1,2…>.

  To see the plumbed interfaces, execute $> ifconfig -a.
- _tx_bcopy_limit - this can be changed at runtime by executing

  ```
  $> dladm set-linkprop -p _tx_bcopy_limit=<value in bytes> <interface>
  ```

  Note: _tx_bcopy_limit is the same as the oce_bcopy_limit parameter in the oce.conf file.

Migrating from the Solaris LPFC Driver to the Solaris emlxs Driver

If the Emulex LPFC driver for Solaris is already installed, you can migrate to the Emulex emlxs driver either by customizing and running the unsupported sample scripts provided by Emulex, or by manually performing a set of procedures. These scripts are available on the SFS driver pages on the Emulex Web site.

Operational Behaviors of the emlxs Driver

- Device Discovery:
  - Device masking in the emlxs driver is managed using the Solaris cfgadm utility.
- Firmware download:
  - Sun-branded adapters: the emlxs driver includes the adapter firmware and overrides any firmware version previously residing on the adapter. You cannot update the firmware manually.
  - Emulex SFS-supported adapters: the Emulex-provided emlxadm tool provides a `download_fw` command. Syntax and details are provided in the *Emulex FCA Utilities Reference Manual*. You can also use the HBAnyware configuration utility. Refer to the *HBAnyware User Manual* for more information.
- Universal Boot download, including OpenBoot (FCode):
  - Sun-branded 2 Gb/s adapters: use Sun-provided luxadm. You can also use the HBAnyware configuration utility. Refer to the *HBAnyware User Manual* for more information.
  - Sun-branded 4 Gb/s adapters: use the Emulex-provided emlxadm tool, which provides a `download_fcode` command (syntax and details are provided in the *Emulex FCA Utilities Reference Manual*). You can also use the HBAnyware configuration utility. Refer to the *HBAnyware User Manual* for more information.

- Emulex SFS-supported 2 Gb/s adapters: use either luxadm or emlxadm. You can also use the HBAnyware configuration utility. Refer to the *HBAnyware User Manual* for more information.
- Emulex SFS-supported 4 Gb/s adapters: use emlxadm. You can also use the HBAnyware configuration utility. Refer to the *HBAnyware User Manual* for more information.

Use Cases

Note: The concurrent production use of emlxs and LPFC on a single server is not supported. Transient co-existence is required in some migration use cases but must be discontinued before going into production.

Different use cases will result in different migration scenarios.

Table 2: Use Cases

Server Platform	Existing LPFC Configuration	Targeted FC Environment	See Section...
x64 and x86	Not applicable	All cases	
SPARC	Existing LPFC driver, no FC boot	emlxs no FC boot	See “Migrating a Configuration without FC Boot” on page 23
	Existing LPFC driver, FC boot	emlxs with FC boot	See “Migrating a Configuration with FC Boot” on page 25

The unsupported sample migration scripts include support for migration in Sun Cluster environments.

This revision does not cover migration of a boot drive, or of logical unit numbers (LUNs) accessed through multipathing software such as EMC PowerPath or Veritas DMP, or of volume managers such as Sun SVM or Veritas VxVM.

Sample Script File Details

Emulex provides unsupported sample scripts to help you migrate from the Solaris LPFC driver to the Solaris emlxs driver. These scripts are available on the SFS driver pages on the Emulex Web site. You can customize these scripts and run them to automate the migration process.

start_emlxs_migration.sh

The `start_emlxs_migration.sh` sample script performs the following tasks:

1. Verifies required packages are installed (3 packages - LPFC driver, emlxs driver and the HBAnyware utility).
2. Cleans up any device-dangling links by running the operating system utility: `devfsadm -C`.
3. Obtains and saves the following information for each adapter in the system:
   - OS device name for the adapter (i.e. reflects PCI path).
   - OS logical controller number for the adapter.
   - Obtains a target number and wwn for all targets configured for each adapter and obtains the number of Luns configured for each target.
4. Writes data to files.

5. Verifies that the system boot device is not an Emulex adapter (if so, the sample script exits with an explanation).
6. Obtains and verifies the FCode version for each adapter in the system. If the FCode version is not compatible, the sample script errors, then exits.
7. Sets the FCode SFS bit to `1` on each adapter.
8. Calls the operating system's add/remove driver utility to configure `/etc/driver_aliases`.
9. Prompts you to reboot the system.

The `adapterN.migrate` and `targetN.migrate` files are generated by the start sample script. These files verify the migration process. Only attached and operational targets are migrated.

- `adapterN.migrate` - where *N* is the adapter number (one file for each adapter); primarily this file contains the adapter device path/name to link LPFC adapters to emlxs adapters across a reboot.
- `targetN.migrate` - where *N* is the adapter number (only adapters with targets configured have this file) - this file has target numbers and WWNs.

finish_emlxs_migration.sh

The finish_emlxs_migration.sh sample script performs the following tasks:

1. Cleans up any device-dangling links by running the operating system utility: `devfsadm -C`.
2. Performs the following tasks for each adapter in the system:
    a. Reads the device name from the file that was generated by `start_emlxs_migration.sh`.
    b. Greps with the `ls -l /dev/cfg` command to acquire the emlxs controller number.
    c. Writes the LPFC controller number and the emlxs controller number to the map file.
    d. Constructs a target device name using the target WWN format (for example, c3::21000004cf92913c) for each target in the target file.
3. Uses the `cfgadm -al` command to grep the target device name output and determine if the target device is already configured. Configures the device if necessary with the `cfgadm -c configure` command.
4. Greps with the `/etc/vfstab` command and replaces any LPFC-based storage device entry with its new emlxs-based storage device name entry using the target WWN device name format (for example, c3::21000004cf92913c).
5. Executes a `mountall -l` if any LPFC storage device entry has been replaced with a new SFS storage device name. Forces the operating system to re-mount local devices with `/etc/vfstab` command.

The `controllermap.migrate` and the `lpfccontroller.migrate` are map files that are generated by the finish sample script.

- `controllermap.migrate` - a file with entries that map the LPFC controller number to the emlxs controller number.
- `lpfccontroller.migrate` - one file with entries that map the adapter LPFC controller numbers to the LPFC adapter numbers (for /etc/vfstab parsing).

Migrating a Configuration without FC Boot

Migrating Automatically

Automatic migration provides an equivalent FC storage setup running on the Solaris FC stack. Emulex's Solaris LPFC driver on the SPARC platform uses "sd" as the native SCSI driver, and works in Solaris 10. Emulex’s emlxs driver supports the Solaris FC stack using "ssd" as the SCSI driver. With this procedure, a SAN setup on the host seamlessly migrates from LPFC to the same setup using emlxs.

Prerequisites

- SPARC server running Solaris 10.
- Emulex's LPFC driver and associated application kit including HBAnyware installed on the host system.
- Emulex's emlxs driver (SUNWemlxs) installed on the host system.
- Emulex FCode version 1.50a4 or later pre-installed on all adapters.

Things to Know Before You Migrate

- FC tape devices do not migrate to the emlxs environment. Configure devices after migration.
- LPFC.conf properties do not migrate into the emlxs driver environment. Note custom configuration values before migration, as default properties are used after migration. Customize applicable properties after the migration completes.
- The Solaris FC stack does not support LUN-level masking. Verify that the system is properly configured to provide the same number of LUNs in emlxs as are contained in the original LPFC environment. For a specific target, any visible LUNs that are not configured in the LPFC environment are automatically configured into the emlxs environment.
- The Solaris FC stack natively supports mpxio. If you use multipathing or load balancing software, verify that the software functions properly in the new emlxs environment.

Limitations

- If an Emulex adapter is the boot adapter, the sample script exits without proceeding with migration.
- If an Emulex adapter is configured to use the IP over FC interface, the IP interface does not migrate to the emlxs environment.

Procedures

To automatically migrate from LPFC to emlxs:

1. Download the migrate .tar file to the host system in which the LPFC driver is in control and untar it. The .tar file contains two sample script files and a subdirectory containing binary files that are used by the sample scripts.
2. Open the .tar file and view the `start_emlxs_migration.sh` and `finish_emlxs_migration.sh` sample script files. Make changes to these files as needed based upon your system configuration.
3. Login as ’root’ and run the `start_emlxs_migration.sh` customized script file (for details, see page 21). After start_emlxs_migration.sh is completed, reboot the host system.
4. Login as ’root’ and change directory (cd) to where the customized migration scripts are installed.
5. Run the `finish_emlxs_migration.sh` customized script file (for details, see page 22).

6. Uninstall HBAnyware for the LPFC driver and install HBAnyware for the emlxs driver.

To manually migrate from LPFC to emlxs:

1. Back up all data and system disks.
2. Note current LPFC target and LUN information contained in the following files:
    - ./etc/vfstab
    - ./kernel/drv/lpfc.conf
    - ./kernel/drv/sd.conf
3. Using Emulex's HBAnyware configuration utility for LPFC (bundled as part of the driver kit available at http://www.emulex.com/support/solaris/index.jsp):
    - Update the FCode in all adapters to the latest version.
    - Verify that FCode is enabled.
4. Uninstall HBAnyware and LPFC as follows:

    Login as ’root’ or su to ’root’, then type `pkgrm HBAnyware lpfc`
5. Install the required emlxs driver package and patch. Follow the instructions in the "Installing the Solaris SFS FCA (emlxs) Driver" section, under *Downloading and Installing the Driver for Solaris 10 and OpenSolaris (SPARC, x64 and x86)* on page 2.
6. Install Emulex's utilities kit for emlxs available at: http://www.emulex.com/support/solaris/sfs_sparc.jsp.
7. Boot to the ok prompt.
8. Issue the Emulex FCode `set-sfs-boot` command to change the Emulex adapter's device path from LPFC to emlx. The change will not take effect until the system is reset.

    Example:

    ```
    {0} ok show-devs
    .
    .
    /pci@8,600000/lpfc@2
    .
    .
    {0} ok " /pci@8,600000/lpfc@2" select-dev
            ^
            Space required
    {0} ok set-sfs-boot
    {0} ok unselect-dev
    ```

    Repeat this step for all adapters in the system. Type `reset-all`, then boot the system to the operating system.
9. Configure any targets that were used with the LPFC driver (`cfgadm -a` to display the target list, `cfgadm -c configure <ApId>` to configure the ApId's storage). The ApId can also be referenced in the /kernel/drv/lpfc.conf file.

    For example, fcp-bind-WWPN="200400a0b816dc52:lpfc3t4" could be configured by typing: `cfgadm -c configure c6::200400a0b816dc52`

10. Edit the /etc/vfstab file and replace the sd pathname (for example, c3t4d1s6) to the ssd pathname (for example, c6t200400A0B816DC52d1s6).

Migrating a Configuration with FC Boot

Prerequisite

- An additional Sun-branded Emulex adapter or Emulex adapter that is supported by emlxs and of a different family from the boot adapter with FCode version 1.50a4 or later and enabled.

Procedure

To manually migrate:

1. Back up all data and system disks.
2. Note current LPFC target and LUN information contained in the following files:
   - /etc/vfstab
   - /kernel/drv/lpfc.conf
   - /kernel/drv/sd.conf.
3. Using Emulex's HBAnyware configuration utility for LPFC (bundled as part of the driver kit available at http://www.emulex.com/support/solaris/index.jsp):
   - Update the FCode in all adapters to the latest version.
   - Verify that FCode is enabled.
4. Install the required emlxs driver package and patch. Follow the instructions in the "Installing the Solaris SFS FCA (emlxs) Driver" section, under *Downloading and Installing the Driver for Solaris 10 and OpenSolaris (SPARC, x64 and x86)* on page 2.
5. Shut down the system with the `shutdown` command.
6. Install the new adapter.
7. Boot the OS.

   If you are migrating the boot adapter to an Emulex adapter, perform the following additional steps. Otherwise skip to step 8.

   a. Identify the device path of the new boot drive, using the following format:

```
# format
.
.
.
   /pci@8,600000/lpfc@2/sd@1,0
.
.
.
```

   b. Use emlxdrv to change only the migrating LPFC-attached adapter family to emlxs. Do not migrate the boot LPFC adapter's family.

   c. Shutdown then boot the system to the ok prompt.

   d. Issue the `set-sfs-boot` command to change the migrated Emulex adapter device paths from LPFC to emlxs:

```
{0} ok show-devs
.
.
.
/pci@8,600000/lpfc@2
.
.
.
{0} ok " /pci@8,600000/lpfc@2" select-dev
        ^
        Space required
{0} ok set-sfs-boot
{0} ok unselect-dev
```

Repeat this step for each of the migrated adapters in the system. Type `reset-all`, then boot the system to the OS.

8. Define or designate an alternate boot drive for DAS boot through SFS and the Sun or Emulex adapter. If the alternate boot drive is fabric-attached, configure the storage (by using a command such as `cfgadm -c configure <ApId>`).
9. Use the `format` command to identify the alternate boot drive and take note of its path because it will be used to boot from the added adapter.
10. Use the `ufsdump` and `ufsrestore` commands to create a fabric boot disk. Follow the instructions until complete. (Refer to the *Emulex Remote Boot Guide for SFS Drivers* at http://www.emulex.com/emc/support/solsfs10_sparc.jsp for more information.)
11. Shut down the server and boot to the ok prompt.
12. Issue the Emulex FCode `set-sfs-boot` command to change the remaining Emulex adapters device paths from LPFC to emlxs. Changes will not take effect until the system is reset.

Example:

```
{0} ok show-devs
.
.
.
/pci@8,600000/lpfc@2
.
.
.
{0} ok " /pci@8,600000/lpfc@2" select-dev
        ^
        Space required
{0} ok set-sfs-boot
{0} ok unselect-dev
```

Repeat this step for each of the remaining adapters in the system. Type `reset-all`, then boot the system to the OS.

13. Boot the new device:
    - For a Sun-branded boot adapter:

      ```
      {0} ok boot
      /pci@8,600000/SUNW,emlxs@2/fp@0,0/disk@w21000004cf720664,0:a
      ```

    - For an Emulex boot adapter:

      ```
      {0} ok boot /pci@8,600000/emlx@2/fp0,0/disk@w21000004cf720664,0:a
      ```

14. Use emlxdrv to migrate the remaining LPFC-attached adapters to emlxs.
15. Configure any targets that were used with the LPFC driver (`cfgadm -a` to display the target list, `cfgadm -c configure <ApId>` to configure the ApId's storage). The ApId can also be referenced in the /kernel/drv/lpfc.conf file.

    For example, fcp-bind-WWPN="200400a0b816dc52:lpfc3t4" could be configured by typing:
    `cfgadm -c configure c6::200400a0b816dc52`
16. Edit the /etc/vfstab file and replace the sd pathname (for example, c3t4d1s6) to the ssd pathname (for example, c6t200400A0B816DC52d1s6).
17. Uninstall the HBAnyware utility and the LPFC driver as follows:

    Login as ’root’ or su to ’root’, then type `pkgrm HBAnyware lpfc`
18. Install Emulex's utilities kit for emlxs available at:
    http://www.emulex.com/support/solaris/sfs_sparc.jsp.

Solaris emlxs and Solaris LPFC Driver Properties Cross-Reference Table

The cross-reference information listed in Table 3 refers to the driver for Solaris LPFC version 6.30.

Table 3: Solaris emlxs and Solaris LPFC Driver Property Cross-Reference

Solaris emlxs/ HBAnyware Property	Solaris emlxs/HBAnyware Min/Max, Defaults and Description	Related LPFC Property	LPFC Min/Max, Default and Description	Comments
ack0	0 = The driver will use ACK1 for class 2 acknowledgement. 1 = The driver will use ACK0 for class 2 acknowledgement. Range: Min:0 Max:1 Default:0 Description: Determines if ACK0 is used instead of ACK1 for class 2 acknowledgement.	ack0	0 = The driver will use ACK1 for class 2 acknowledgement. 1 = The driver will use ACK0 for class 2 acknowledgement. Range: Min:0 Max:1 Default:0 Description: Determines if ACK0 is used instead of ACK1 for class 2 acknowledgement.	
adisc-support	0 = No support. Flush active I/O's for all FCP target devices at link down. 1 = Partial support. Flush I/O's for non-FCP2 target devices at link down. 2 = Full support. Hold active I/O's for all devices at link down. Default: 1 Description: Sets the level of driver support for the FC ADISC login I/O recovery method.	use-adisc	0 = Off 1 = On Default: 0 Description: Controls the ELS command used for address authentication during rediscovery upon link-up. The driver will always use ADISC for FCP-2 devices and re-discovery due to an registered state change notification (RSCN).	If there are tape devices on the SAN that support FCP2, set the use-adisc property to 1 and the adisc-support property to 1 (partial support) or 2 (full support).
assign-alpa	Min:0x00 Max:0xef Default:0x00 (valid ALPA's only) Description: This is only valid if topology is loop. A zero setting means no preference. If multiple adapter instances on the same host are on the same loop, you should set this value differently for each adapter.	assign-alpa	Description:Sets a preferred ALPA for the adapter. This parameter is only valid if the topology is loop.	

Table 3: Solaris emlxs and Solaris LPFC Driver Property Cross-Reference (Continued)

Solaris emlxs/ HBAnyware Property	Solaris emlxs/HBAnyware Min/Max, Defaults and Description	Related LPFC Property	LPFC Min/Max, Default and Description	Comments
console-notices	Min: 0x00000000 Max:0xFFFFFFFF Default: 0x0000000 Verbose mask for notice messages to the console.	log-verbose	Min:0x0 Max:0xffff Default:0x0 (bit mask) When set to nonzero this variable causes LPFC to generate additional messages concerning the state of the driver and the I/O operations it carries out. These messages may go to the system console. Log-only for lpfc.conf file.	
console-warnings	Min: 0x00000000 Max:0xFFFFFFFF Default: 0x0000000 Verbose mask for warning messages to the console.	log-verbose	Min:0x0 Max:0xffff Default:0x0 (bit mask) When set to nonzero this variable causes LPFC to generate additional messages concerning the state of the driver and the I/O operations it carries out. These messages may go to the system console. Log-only for lpfc.conf file.	
console-errors	Min: 0x00000000 Max:0xFFFFFFFF Default: 0x0000000 Verbose mask for error messages to the console.	log-verbose	Min:0x0 Max:0xffff Default:0x0 (bit mask) When set to nonzero this variable causes LPFC to generate additional messages concerning the state of the driver and the I/O operations it carries out. These messages may go to the system console. Log-only for lpfc.conf file.	
cr-delay	Min:0 Max:63 Default:0 Description: Specifies a count of milliseconds after which an interrupt response is generated if the cr-count has not been satisfied. This value is set to 0 to disable the Coalesce Response feature as default.	cr-delay	Min:0 Max:63 Default:0 Description: Specifies a count of milliseconds after which an interrupt response is generated if the cr-count has not been satisfied. This value is set to 0 to disable the Coalesce Response feature as default.	Setting this value can minimize CPU utilization by reducing the number of interrupts that the driver generates to the operating system.

Table 3: Solaris emlxs and Solaris LPFC Driver Property Cross-Reference (Continued)

Solaris emlxs/ HBAnyware Property	Solaris emlxs/HBAnyware Min/Max, Defaults and Description	Related LPFC Property	LPFC Min/Max, Default and Description	Comments
cr-count	Min:1 Max:255 Default:1 Description: Specifies a count of I/O completions after which an interrupt response is generated. This feature is disabled if cr-delay is set to 0.	cr-count	Min:1 Max:255 Default:1 Description: Specifies a count of I/O completions after which an interrupt response is generated. This feature is disabled if cr-delay is set to 0.	The value is often determined by your OEM. This property sets the number of I/Os to be queued in the operating system's driver before an interrupt is initiated. The driver default settings are roughly a 1:1 I/O to interrupt ratio. If you change this property, performance varies per application.
link-speed	0 = auto select 1 = 1 Gb/S 2 = 2 Gb/S 4 = 4 Gb/S 8 = 8 Gb/S Range: 0 - 8 Default: 0 Description: Sets the link speed setting for initializing the FC connection.	link-speed	0 = auto select 1 = 1 Gb/S 2 = 2 Gb/S 4 = 4 Gb/S 8 = 8 Gb/S Range: 0 - 8 Default: 0 Description: Sets link speed.	This value can be changed to a specific link speed to optimize the link initialization process for a specific environment.
log-notices	Min: 0x00000000 Max:0xFFFFFFFF Default: 0x0000000 Verbose mask for notice messages to the messages file.	log-verbose	Min:0x0 Max:0xffff Default:0x0 (bit mask) When set to nonzero this variable causes LPFC to generate additional messages concerning the state of the driver and the I/O operations it carries out. These messages may go to the system log file, /var/adm/messages.	

Table 3: Solaris emlxs and Solaris LPFC Driver Property Cross-Reference (Continued)

Solaris emlxs/ HBAnyware Property	Solaris emlxs/HBAnyware Min/Max, Defaults and Description	Related LPFC Property	LPFC Min/Max, Default and Description	Comments
log-warnings	Min: 0x00000000 Max:0xFFFFFFFF Default: 0x0000000 Verbose mask for warning messages to the messages file.	log-verbose	Min:0x0 Max:0xffff Default:0x0 (bit mask) When set to nonzero this variable causes LPFC to generate additional messages concerning the state of the driver and the I/O operations it carries out. These messages may go to the system log file, /var/adm/messages.	
log-errors	Min: 0x00000000 Max:0xFFFFFFFF Default: 0x0000000 Verbose mask for error messages to the messages file.	log-verbose	Min:0x0 Max:0xffff Default:0x0 (bit mask) When set to nonzero this variable causes LPFC to generate additional messages concerning the state of the driver and the I/O operations it carries out. These messages may go to the system log file, /var/adm/messages.	
max-xfer-size	Min: 131072 Max:1388544 Default: 339968 Determines the scatter gather list buffer size. A pool of buffers is reallocated by the driver during boot. A larger transfer size requires a larger memory allocation.	N/A		
network-on	Min:0 (Disables) Max:1 (Enables) Default:1 Description: Enables or disables IP networking support in the driver.	network-on	Min:0 (Disables) Max:1 (Enables) Default: 0 Description: Controls whether LPFC provides IP networking functionality over FC. This variable is Boolean: when zero, IP networking is disabled: when non-zero, IP networking is enabled. This variable is set during the installation of the driver via pkgadd.	

Table 3: Solaris emlxs and Solaris LPFC Driver Property Cross-Reference (Continued)

Solaris emlxs/ HBAnyware Property	Solaris emlxs/HBAnyware Min/Max, Defaults and Description	Related LPFC Property	LPFC Min/Max, Default and Description	Comments
num-iocbs	Min:128 Max:10240 Default = 1024 Description: Sets the number of iocb buffers to allocate.	num-iocbs	Min:128 Max:10240 Default = 256 Description: Specifies the number of command buffers to allocate. These buffers are used for Fibre Channel Extended Link Services (ELS) and one for each FCP command issued in SLI-2 mode. If you want to queue lots of FCP commands to the adapter, then you should increase num-bufs for better performance. These buffers consume physical memory and are also used by the device driver to process loop initialization and rediscovery activities. **Important:** The driver must always be configured with at least several dozen ELS command buffers; Emulex recommends at least 128.	
num-nodes	Min:2 Max:512 Default:512 Description: Number of FC nodes (NPorts) the driver will support.	N/A	N/A	
pci-max-read	Min: 512 Max: 4092 Default: 2048 Description: Sets the PCI-X max memory read byte count [512, 1024, 2048 or 4096].	N/A	N/A	
pm-support	0 = Disables power management support in the driver. 1 = Enables power management support in the driver. Default: 0 Description: Enable/Disable power management support in the driver	N/A	N/A	

Table 3: Solaris emlxs and Solaris LPFC Driver Property Cross-Reference (Continued)

Solaris emlxs/ HBAnyware Property	Solaris emlxs/HBAnyware Min/Max, Defaults and Description	Related LPFC Property	LPFC Min/Max, Default and Description	Comments
topology	0 = loop, if it fails attempt pt-to-pt 2 = pt-to-pt only 4 = loop only 6 = pt-to-pt, if it fails attempt loop Default: 0 Description: Link topology for initializing the Fibre Channel connection. Set pt-to-pt if you want to run as an N_Port. Set loop if you want to run as an NL_Port.	topology	0x0 = loop, if it fails attempt pt-to-pt 0x2 = pt-to-pt only 0x4 = loop only Default: 0 Description: Controls the FC topology expected by LPFC at boot time. FC offers pt-to-pt, fabric and arbitrated loop topologies. To make the adapter operate as an N_Port, select pt-to-pt mode (used for N_Port to F_Port and N_Port to N_Port connections). To make the adapter operate as an NL_Port, select loop mode (used for private loop and public loop topologies). The driver will reject an attempt to set the topology to a value not in the above list. The auto-topology settings 0 and 6 will not work unless the adapter is using firmware version 3.20 or higher.	The topology property controls the protocol (not physical) topology attempted by the driver.
ub-bufs	Min:40 Max:16320 Default:1000 Description: Sets the number of unsolicited buffers to be allocated.	N/A	N/A	

Console and Log Messages

emlxs Logs

General Situations

If an FC link fails to come up, verify that an 8 Gb/s adapter is not attempting to connect to a 1 Gb/s device. Only 2 Gb/s, 4 Gb/s and 8 Gb/s devices are supported on 8 Gb/s adapters.

Messages

This section describes the type of console and log messages you may see. Security levels and an extensive listing of message IDs and descriptions are also provided. Log messages are logged to the /var/adm/messages system file.

Table 4 lists the types of notices, warnings and error logging levels you may set.

Table 4: Notice, Warnings and Error Types

Driver Property	Default/ Min/Max	Effect of Changing Default	Related lpfc Driver Property
console-notices	0	Sets the verbose level for driver notices to the console.	log-only (when set to 0, log messages are logged to the system log file and also printed on the console.) Default = Disabled
console-warnings	0	Sets the verbose level for driver warnings to the console.	
console-errors	0	Sets the verbose level for driver errors to the console.	
log-notices	0xffffffff;	Sets the verbose level for driver notices to the system log file.	log-verbose (when set to non-zero, verbose messages are generated.) Default = Disabled
log-warnings	0xffffffff;	Sets the verbose level for driver warnings to the system log file.	
log-errors	0xffffffff;	Sets the verbose level for driver errors to the system log file.	

Table 5 lists the types of log messages that can be logged to the system file.

Table 5: Log Message Types

LOG Message Verbose Mask	Verbose Bit	Verbose Description
LOG_MISC	0x00000001	Miscellaneous events
LOG_DRIVER	0x00000002	Driver attach and detach events

Table 5: Log Message Types (Continued)

LOG Message Verbose Mask	Verbose Bit	Verbose Description
LOG_INIT	0x00000004	HBA Initialization events
LOG_MEM	0x00000008	Memory management events
LOG_SLI	0x00000010	Service Level Interface (SLI) events
LOG_MBOX	0x00000020	Mailbox events
LOG_NODE	0x00000040	Node events
LOG_LINK	0x00000080	Link events
LOG_ELS	0x00000100	ELS events
LOG_PKT	0x00000200	General I/O packet events
LOG_FCP	0x00000400	FCP traffic events
LOG_FCT	0x00000800	FCP target mode events
LOG_IP	0x00001000	IP traffic events
LOG_SFS	0x00002000	Solaris SFS events
LOG_IOCTL	0x00004000	IOCTL events
LOG_FIRMWARE	0x00008000	Firmware download events
LOG_CT	0x00010000	CT events
LOG_FCSP	0x00020000	FCSP events
LOG_RESERVED	0x007C0000	Reserved for future use
LOG_FCT_DETAIL	0x00800000	Detailed FCT events
LOG_FCSP_DETAIL	0x01000000	Detailed FCSP events
LOG_NODE_DETAIL	0x02000000	Detailed node events
LOG_IOCTL_DETAIL	0x04000000	Detailed IOCTL events
LOG_IP_DETAIL	0x08000000	Detailed IP events
LOG_FIRMWARE_DETAIL	0x10000000	Detailed Firmware events
LOG_SFS_DETAIL	0x20000000	Detailed Solaris SFS events
LOG_MBOX_DETAIL	0x40000000	Detailed Mailbox events
LOG_SLI_DETAIL	0x80000000	Detailed HBA SLI events
LOG_ALL_MSG	0XFFFFFFFF	Detailed Node events

Severity Levels

Table 6: Severity Levels

Level	Message Description
DEBUG (Informational	Message provides engineering debug information.
NOTICE (Informational)	Message provides a general purpose information.
WARNING	Message provides a general purpose warning.
ERROR	Message indicates that a driver error has occurred.
PANIC (Severe)	Message indicates that the driver has forced a system panic to occur.

Message Log Example

The following is an example of a message on the system console.

```
[5.0336]emlxs0: NOTICE: 720: Link up. (1Gb, fabric)
```

The following is an example of the same message in the system message log (/var/adm/messages) file.

```
Jan 19 14:45:36 sunv240 emlxs: [ID 349649 kern.info] [5.0336]emlxs0: NOTICE: 720: Link up. (1Gb, 
fabric)
```

In the above system log message:

- Jan 19 14:45:36 unidentified the date and time when the error or event occurred.
- sunv240 identifies the name of the host machine.
- emlxs identifies the message came from the Emulex emlxs driver.
- [ID 349649 kern.info] identifies a Solaris-specific message ID and kernel message level. This will change from one driver message to another.
- [5.0336] identifies the emlxs driver message context tag. This may change from one driver version to another.
- emlxs0 identifies the message is coming from the emlxs driver instance zero. This will change from one driver instance to another.
- NOTICE identifies the emlxs message severity level. This may change from one driver version to another.
- 720 identifies the emlxs drive message id. This will not change from one driver version to another.
- Link up identifies the actual error or event message. This will not change from one driver version to another.
- (1 Gb/s, fabric) identifies additional information specific to the error or event message. This information is normally intended for technical support / engineering use. This may change from one driver version to another.

Miscellaneous Events

MSG_ID: 0001 DEBUG:

VERBOSE_MASK: LOG_MISC (0x00000001)
DESCRIPTION: This is a general purpose informational message.
SEVERITY LEVEL: Debug
MESSAGE: None
ACTION: No action needed, informational.

MSG_ID: 0002 NOTICE:

VERBOSE_MASK: LOG_MISC (0x00000001)
DESCRIPTION: This is a general purpose informational message.
SEVERITY LEVEL: Notice
MESSAGE: None
ACTION: No action needed, informational.

MSG_ID: 0003 WARNING:

VERBOSE_MASK: LOG_MISC (0x00000001)
DESCRIPTION: This is a general purpose warning message.
SEVERITY LEVEL: Warning
MESSAGE: None
ACTION: No action needed, informational.

MSG_ID: 0004 ERROR:

VERBOSE_MASK: LOG_MISC (0x00000001)
DESCRIPTION: This is a general purpose error message.
SEVERITY LEVEL: Error
MESSAGE: None
ACTION: No action needed, informational.

MSG_ID: 0005 PANIC:

VERBOSE_MASK: LOG_MISC (0x00000001)
DESCRIPTION: This is a general purpose panic message.
SEVERITY LEVEL: Panic (Severe)
MESSAGE: None
ACTION: Contact your customer service representative.

MSG_ID: 0010 DEBUG: Event.

VERBOSE_MASK: LOG_MISC (0x00000001)
DESCRIPTION: This is debug information about a driver event.
SEVERITY LEVEL: Debug
MESSAGE: Event.
ACTION: No action needed, informational.

MSG_ID: 0011 DEBUG: Event queued.

VERBOSE_MASK: LOG_MISC (0x00000001)
DESCRIPTION: This indicates a driver event is being queued.
SEVERITY LEVEL: Debug
MESSAGE: Event queued.
ACTION: No action needed, informational.

MSG_ID: 0012 DEBUG: Event dequeued.

VERBOSE_MASK: LOG_MISC (0x00000001)
DESCRIPTION: This indicates a driver event is being dequeued.
SEVERITY LEVEL: Debug
MESSAGE: Event dequeued.
ACTION: No action needed, informational.

Driver Events

MSG_ID: 0100 NOTICE: Driver attach.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver is performing an attach operation.
SEVERITY LEVEL: Notice
MESSAGE: Driver attach.
ACTION: No action needed, informational.

MSG_ID: 0101 ERROR: Driver attach failed.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver was unable to attach due to some issue.
SEVERITY LEVEL: Error
MESSAGE: Driver attach failed.
ACTION: Check your hardware and software configuration. If the problem persists, report these errors to your customer service representative.

MSG_ID: 0102 DEBUG: Driver attach.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver is performing a attach operation.
SEVERITY LEVEL: Debug
MESSAGE: Driver attach.
ACTION: No action needed, informational.

MSG_ID: 0110 NOTICE: Driver detach.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver is performing a detach operation.
SEVERITY LEVEL: Notice
MESSAGE: Driver detach.
ACTION: No action needed, informational.

MSG_ID: 0111 ERROR: Driver detach failed.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver was unable to detach due to some issue.
SEVERITY LEVEL: Error
MESSAGE: Driver detach failed.
ACTION: Check your hardware and software configuration. If the problem persists, report these errors to your customer service representative.

MSG_ID: 0112 DEBUG: Driver detach.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver is performing a detach operation.
SEVERITY LEVEL: Debug
MESSAGE: Driver detach.
ACTION: No action needed, informational.

MSG_ID: 0120 DEBUG: Driver suspend.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver is performing a suspend operation.
SEVERITY LEVEL: Debug
MESSAGE: Driver suspend.
ACTION: No action needed, informational.

MSG_ID: 0121 ERROR: Driver suspend failed.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver was unable to suspend due to some issue.
SEVERITY LEVEL: Error
MESSAGE: Driver suspend failed.
ACTION: Check your hardware and software configuration. If the problem persists, report these errors to your customer service representative.

MSG_ID: 0130 DEBUG: Driver resume.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver is performing a resume operation.
SEVERITY LEVEL: Debug
MESSAGE: Driver resume.
ACTION: No action needed, informational.

MSG_ID: 0131 ERROR: Driver resume failed.

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This indicates that the driver was unable to resume due to some issue.
SEVERITY LEVEL: Error
MESSAGE: Driver resume failed.
ACTION: Check your hardware and software configuration. If the problem persists, report these errors to your customer service representative.

HBA Initialization Events

MSG_ID: 0200 NOTICE: Adapter initialization.

VERBOSE_MASK: LOG_INIT (0x00000004)
DESCRIPTION: This indicates that the adapter is initializing.
SEVERITY LEVEL: Notice
MESSAGE: Adapter initialization.
ACTION: No action needed, informational.

MSG_ID: 0201 ERROR: Adapter initialization failed.

VERBOSE_MASK: LOG_INIT (0x00000004)
DESCRIPTION: This indicates that an attempt to initialize the adapter has failed.
SEVERITY LEVEL: Error
MESSAGE: Adapter initialization failed.
ACTION: Check your hardware configuration. If the problem persists, report these errors to your customer service representative.

MSG_ID: 0202 DEBUG: Adapter initialization.

VERBOSE_MASK: LOG_INIT (0x00000004)
DESCRIPTION: This indicates that the adapter is initializing.
SEVERITY LEVEL: Debug
MESSAGE: Adapter initialization.
ACTION: No action needed, informational.

MSG_ID: 0210 DEBUG: Adapter transition.

VERBOSE_MASK: LOG_INIT (0x00000004)
DESCRIPTION: This indicates that the adapter is changing states.
SEVERITY LEVEL: Debug
MESSAGE: Adapter transition.
ACTION: No action needed, informational.

MSG_ID: 0220 DEBUG: Adapter online.

VERBOSE_MASK: LOG_INIT (0x00000004)
DESCRIPTION: This indicates that the adapter is online and ready to communicate.
SEVERITY LEVEL: Debug
MESSAGE: Adapter online.
ACTION: No action needed, informational.

MSG_ID: 0230 DEBUG: Adapter offline.

VERBOSE_MASK: LOG_INIT (0x00000004)
DESCRIPTION: This indicates that the adapter is offline and unable to communicate.
SEVERITY LEVEL: Debug
MESSAGE: Adapter offline.
ACTION: No action needed, informational.

MSG_ID: 0231 WARNING: Adapter shutdown.

VERBOSE_MASK: LOG_INIT (0x00000004)
DESCRIPTION: This indicates that the adapter has been shutdown and will require a reboot to reinitialize.
SEVERITY LEVEL: Warning
MESSAGE: Adapter shutdown.
ACTION: Contact your customer service representative.

MSG_ID: 0240 ERROR: Adapter reset failed.

VERBOSE_MASK: LOG_INIT (0x00000004)
DESCRIPTION: This indicates that an attempt to reset the adapter has failed.
SEVERITY LEVEL: Error
MESSAGE: Adapter reset failed.
ACTION: Check your hardware configuration. If the problem persists, report these errors to your customer service representative.

Memory Management Events

MSG_ID: 0300 DEBUG: Memory alloc.

VERBOSE_MASK: LOG_MEM (0x00000008)
DESCRIPTION: This indicates that the driver allocated system memory.
SEVERITY LEVEL: Debug
MESSAGE: Memory alloc.
ACTION: No action needed, informational.

MSG_ID: 0301 ERROR: Memory alloc failed.

VERBOSE_MASK: LOG_MEM (0x00000008)
DESCRIPTION: This indicates that the driver was unable to allocate system memory. The system is low on memory resources.
SEVERITY LEVEL: Error
MESSAGE: Memory alloc failed.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your system administrator.

MSG_ID: 0310 ERROR: Memory pool error.

VERBOSE_MASK: LOG_MEM (0x00000008)
DESCRIPTION: This indicates that a problem has occurred with the memory buffer pool management.
SEVERITY LEVEL: Error
MESSAGE: Memory pool error.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0311 DEBUG: Memory pool alloc failed.

VERBOSE_MASK: LOG_MEM (0x00000008)
DESCRIPTION: This indicates that the driver was unable to allocate memory from one of its own memory pools.
SEVERITY LEVEL: Debug
MESSAGE: Memory pool alloc failed.
ACTION: If the problem occurs frequently you may be able to configure more resources for that pool. If this

does not solve the problem, report these errors to customer service.

MSG_ID: 0312 DEBUG: Memory pool detail.

VERBOSE_MASK: LOG_MEM (0x00000008)
DESCRIPTION: This provides detailed information about memory bufferpool management.
SEVERITY LEVEL: Debug
MESSAGE: Memory pool detail.
ACTION: No action needed, informational.

MSG_ID: 0320 NOTICE: No unsolicited buffer available.

VERBOSE_MASK: LOG_MEM (0x00000008)
DESCRIPTION: This indicates that the driver's unsolicited buffer pool is exhausted. The I/O will be dropped and most likely retried by the remote device.
SEVERITY LEVEL: Notice
MESSAGE: No unsolicited buffer available.
ACTION: If the problem occurs frequently you may be able to configure more resources for that pool. If this does not solve the problem, report these errors to customer service.

MSG_ID: 0330 ERROR: Invalid access handle.

VERBOSE_MASK: LOG_MEM (0x00000008)
DESCRIPTION: This indicates that the driver had an invalid access handle assigned by the system.
SEVERITY LEVEL: Error
MESSAGE: Invalid access handle.
ACTION: If the problem occurs frequently, report these errors to customer service.

MSG_ID: 0331 ERROR: Invalid DMA handle.

VERBOSE_MASK: LOG_MEM (0x00000008)
DESCRIPTION: This indicates that the driver had an invalid dma handle assigned by the system.
SEVERITY LEVEL: Error
MESSAGE: Invalid DMA handle.
ACTION: If the problem occurs frequently, report these errors to customer service.

Service Level Interface (SLI) Events

MSG_ID: 0400 DEBUG: Vital Product Data.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This provides vendor specific information about the adapter.
SEVERITY LEVEL: Debug
MESSAGE: Vital Product Data.
ACTION: No action needed, informational.

MSG_ID: 0410 DEBUG: Link atten.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that the adapter has triggered a link attention interrupt.
SEVERITY LEVEL: Debug
MESSAGE: Link atten.
ACTION: No action needed, informational.

MSG_ID: 0411 DEBUG: State change.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that the adapter has changed state.
SEVERITY LEVEL: Debug
MESSAGE: State change.
ACTION: No action needed, informational.

MSG_ID: 0412 DEBUG: Link Up atten.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that the adapter has triggered a link up attention interrupt.
SEVERITY LEVEL: Debug
MESSAGE: Link Up atten.
ACTION: No action needed, informational.

MSG_ID: 0413 DEBUG: Link Down atten.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that the adapter has triggered a link down attention interrupt.
SEVERITY LEVEL: Debug
MESSAGE: Link Down atten.
ACTION: No action needed, informational.

MSG_ID: 0420 ERROR: Adapter hardware error.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that an interrupt has occurred and the status register indicates a nonrecoverable hardware
SEVERITY LEVEL: Error
MESSAGE: Adapter hardware error.
ACTION: error. This error usually indicates a hardware problem with the adapter. Try running adapter diagnostics. Report these errors to customer service.

MSG_ID: 0421 NOTICE: Adapter temperature.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that the adapter has provided general information about the adapter's temperature.
SEVERITY LEVEL: Notice
MESSAGE: Adapter temperature.
ACTION: No action needed, informational.

MSG_ID: 0422 WARNING: Adapter temperature.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that adapter's temperature is too hot.
SEVERITY LEVEL: Warning
MESSAGE: Adapter temperature.
ACTION: Check hardware ventilation. Reduce adapter usage. Shutdown host system.

MSG_ID: 0423 NOTICE: Adapter notice.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that the adapter has provided general information about the adapter's condition.
SEVERITY LEVEL: Notice
MESSAGE: Adapter notice.
ACTION: No action needed, informational.

MSG_ID: 0424 WARNING: Adapter warning.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that an interrupt has occurred indicating a recoverable adapter error.
SEVERITY LEVEL: Warning
MESSAGE: Adapter warning.
ACTION: This error usually indicates a hardware or firmware problem with the adapter. Check and/or update firmware levels. Report these errors to customer service.

MSG_ID: 0425 ERROR: Adapter error.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that a recoverable adapter error has occurred.
SEVERITY LEVEL: Error
MESSAGE: Adapter error.
ACTION: This error usually indicates a hardware or firmware problem with the adapter. Check and/or update firmware levels. Report these errors to customer service.

MSG_ID: 0426 NOTICE: Adapter Async Status.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that the adapter has provided general information about the adapter's async status.
SEVERITY LEVEL: Notice
MESSAGE: Adapter Async Status.
ACTION: No action needed, informational.

MSG_ID: 0430 DEBUG: Ring event.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that an SLI ring event has occurred.
SEVERITY LEVEL: Debug
MESSAGE: Ring event.
ACTION: No action needed, informational.

MSG_ID: 0431 DEBUG: Ring error.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates an SLI ring error is being reported by the adapter
SEVERITY LEVEL: Debug
MESSAGE: Ring error.
ACTION: No action needed, informational.

MSG_ID: 0432 DEBUG: Ring reset.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates an SLI ring is being reset.
SEVERITY LEVEL: Debug
MESSAGE: Ring reset.
ACTION: No action needed, informational.

MSG_ID: 0440 DEBUG: Adapter msg.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that a message was sent to the driver from the adapter.
SEVERITY LEVEL: Debug
MESSAGE: Adapter msg.
ACTION: No action needed, informational.

MSG_ID: 0450 ERROR: IOCB invalid.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that an IOCB was received from the adapter with an illegal value. This error could indicate a driver or firmware problem.
SEVERITY LEVEL: Error
MESSAGE: IOCB invalid.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0451 DEBUG: IOCB queue full.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates that the IOCB queue is full. This will occur during normal operation.
SEVERITY LEVEL: Debug
MESSAGE: IOCB queue full.
ACTION: No action needed, informational.

MSG_ID: 0452 DEBUG: IOCB event.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates an IOCB local error event is being reported by the adapter
SEVERITY LEVEL: Debug
MESSAGE: IOCB event.
ACTION: No action needed, informational.

MSG_ID: 0453 DEBUG: IOCB stale.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This indicates an IOCB completed after its associated packet completed.
SEVERITY LEVEL: Debug
MESSAGE: IOCB stale.
ACTION: No action needed, informational.

MSG_ID: 0460 DEBUG: SLI detail.

VERBOSE_MASK: LOG_SLI_DETAIL (0x80000000)
DESCRIPTION: This provides detailed information about an SLI event.
SEVERITY LEVEL: Debug
MESSAGE: SLI detail.
ACTION: No action needed, informational.

MSG_ID: 0461 ERROR: SLI ERROR.

VERBOSE_MASK: LOG_SLI (0x00000010)
DESCRIPTION: This provides error information about an SLI event.
SEVERITY LEVEL: Error
MESSAGE: SLI ERROR.
ACTION: No action needed, informational.

Mailbox Events

MSG_ID: 0500 DEBUG: Mailbox event.

VERBOSE_MASK: LOG_MBOX (0x00000020)
DESCRIPTION: This indicates that a mailbox event has occurred.
SEVERITY LEVEL: Debug
MESSAGE: Mailbox event.
ACTION: No action needed, informational.

MSG_ID: 0501 DEBUG: Mailbox detail.

VERBOSE_MASK: LOG_MBOX_DETAIL (0x40000000)
DESCRIPTION: This provides detailed information about a mailbox event.
SEVERITY LEVEL: Debug
MESSAGE: Mailbox detail.
ACTION: No action needed, informational.

MSG_ID: 0510 DEBUG: Stray mailbox interrupt.

VERBOSE_MASK: LOG_MBOX (0x00000020)
DESCRIPTION: This indicates that a mailbox command completion interrupt was received and the mailbox is not valid. This error could indicate a driver or firmware problem.
SEVERITY LEVEL: Debug
MESSAGE: Stray mailbox interrupt.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0520 DEBUG: Mailbox error.

VERBOSE_MASK: LOG_MBOX (0x00000020)
DESCRIPTION: This indicates that an unsupported or illegal mailbox command was completed. This error could indicate a driver or firmware problem.
SEVERITY LEVEL: Debug
MESSAGE: Mailbox error.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0530 ERROR: Mailbox timeout.

VERBOSE_MASK: LOG_MBOX (0x00000020)
DESCRIPTION: The firmware did not response a mailbox command. This error could indicate a hardware or firmware problem.
SEVERITY LEVEL: Error
MESSAGE: Mailbox timeout.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

Node Events

MSG_ID: 0600 DEBUG: Node create.

VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that a node has been created for a remote device.
SEVERITY LEVEL: Debug
MESSAGE: Node create.
ACTION: No action needed, informational.

MSG_ID: 0601 DEBUG: Node opened.

VERBOSE_MASK: LOG_NODE_DETAIL (0x02000000)
DESCRIPTION: This indicates that a node has been opened for IO transport.
SEVERITY LEVEL: Debug
MESSAGE: Node opened.
ACTION: No action needed, informational.

MSG_ID: 0602 NOTICE: Node create failed.

VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that a node create request for a remote device has failed.
SEVERITY LEVEL: Notice
MESSAGE: Node create failed.
ACTION: No action needed, informational.

MSG_ID: 0603 DEBUG: Node updated.

VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that a node has been updated for a remote device.
SEVERITY LEVEL: Debug
MESSAGE: Node updated.
ACTION: No action needed, informational.

MSG_ID: 0610 DEBUG: Node destroy.

VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that a node has been destroyed for a remote device.
SEVERITY LEVEL: Debug
MESSAGE: Node destroy.
ACTION: No action needed, informational.

MSG_ID: 0611 DEBUG: Node closed.

VERBOSE_MASK: LOG_NODE_DETAIL (0x02000000)
DESCRIPTION: This indicates that a node has been temporarily closed for IO transport.
SEVERITY LEVEL: Debug
MESSAGE: Node closed.
ACTION: No action needed, informational.

MSG_ID: 0612 NOTICE: Node missing.

VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that a FCP2 device node has been found missing.
SEVERITY LEVEL: Notice
MESSAGE: Node missing.
ACTION: No action needed, informational.

MSG_ID: 0620 DEBUG: Node not found.

VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that there was an attempt to send an I/O pkt to an unknown device node. The driver maintains a node table entry for every device it needs to communicate with on the FC network.
SEVERITY LEVEL: Debug
MESSAGE: Node not found.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0621 DEBUG: Node timeout.

VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that the node timer expired. This means the node is ready to be opened or it has been offline too long and needs to be flushed.
SEVERITY LEVEL: Debug
MESSAGE: Node timeout.
ACTION: No action needed, informational.

Link Events

MSG_ID: 0700 DEBUG: Link event.

VERBOSE_MASK: LOG_SLI (0x00000010) or LOG_LINK (0x00000080)
DESCRIPTION: This indicates that a link event has occurred.
SEVERITY LEVEL: Debug
MESSAGE: Link event.
ACTION: No action needed, informational.

MSG_ID: 0710 NOTICE: Link down.

VERBOSE_MASK: LOG_LINK (0x00000080)
DESCRIPTION: This indicates that the fibre channel link is down to the adapter.
SEVERITY LEVEL: Notice
MESSAGE: Link down.
ACTION: Check your network connections. If the problem persists, report these errors to your system administrator.

MSG_ID: 0720 NOTICE: Link up.

VERBOSE_MASK: LOG_LINK (0x00000080)
DESCRIPTION: This indicates that the fibre channel link is up.
SEVERITY LEVEL: Notice
MESSAGE: Link up.
ACTION: No action needed, informational.

MSG_ID: 0721 NOTICE: NPIV Link up.

VERBOSE_MASK: LOG_LINK (0x00000080)
DESCRIPTION: This indicates that the fibre channel link is up for all virtual ports.
SEVERITY LEVEL: Notice
MESSAGE: NPIV Link up.
ACTION: No action needed, informational.

MSG_ID: 0730 NOTICE: Link reset.

VERBOSE_MASK: LOG_LINK (0x00000080) or LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an issue has forced the fibre channel link to be reset.
SEVERITY LEVEL: Notice
MESSAGE: Link reset.
ACTION: No action needed, informational.

MSG_ID: 0731 ERROR: Link reset failed.

VERBOSE_MASK: LOG_LINK (0x00000080) or LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an attempt to reset the fibre channel link has failed.
SEVERITY LEVEL: Error
MESSAGE: Link reset failed.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

ELS Events

MSG_ID: 0800 DEBUG: ELS sent.

VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an ELS command is being sent.
SEVERITY LEVEL: Debug
MESSAGE: ELS sent.
ACTION: No action needed, informational.

MSG_ID: 0801 DEBUG: ELS comp.

VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an ELS command completed normally.
SEVERITY LEVEL: Debug
MESSAGE: ELS comp.
ACTION: No action needed, informational.

MSG_ID: 0810 ERROR: Stray ELS completion.

VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that the an ELS command completion was received without issuing a corresponding ELS command. This error could indicate a driver or firmware problem.
SEVERITY LEVEL: Error
MESSAGE: Stray ELS completion.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0811 DEBUG: Abnormal ELS completion.

VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an ELS command completed with a status error in the IOCB. It could mean the Fibre Channel device on the network is not responding or the Fibre Channel device is not an FCP target. The driver will automatically
SEVERITY LEVEL: Debug
MESSAGE: Abnormal ELS completion.
ACTION: retry this ELS command if needed. If the command is a PLOGI or PRLI, and the destination PortID is not an FCP Target, no action is needed. Otherwise, check physical connections to Fibre Channel network and the state the remote PortID is in.

MSG_ID: 0820 DEBUG: ELS rcvd.

VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an unsolicited ELS command was received.
SEVERITY LEVEL: Debug
MESSAGE: ELS rcvd.
ACTION: No action needed, informational.

MSG_ID: 0821 DEBUG: Unsolicited ELS dropped.

VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an unsolicited ELS command was received and then dropped for some reason.
SEVERITY LEVEL: Debug
MESSAGE: Unsolicited ELS dropped.
ACTION: No action needed, informational.

MSG_ID: 0822 DEBUG: ELS reply.

VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that a reply is being sent for an unsolicited ELS command.
SEVERITY LEVEL: Debug
MESSAGE: ELS reply.
ACTION: No action needed, informational.

MSG_ID: 0830 ERROR: Invalid ELS command found.

VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an ELS command was found with an invalid command code.
SEVERITY LEVEL: Error
MESSAGE: Invalid ELS command found.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your

customer service representative.

General I/O Packet Events

MSG_ID: 0900 NOTICE: Packet abort.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an I/O packet is being aborted.
SEVERITY LEVEL: Notice
MESSAGE: Packet abort.
ACTION: No action needed, informational.

MSG_ID: 0901 WARNING: Packet abort failed.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an attempt to abort an I/O packet has failed.
SEVERITY LEVEL: Warning
MESSAGE: Packet abort failed.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0910 DEBUG: Packet timeout.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an I/O packet has timed out and is being aborted.
SEVERITY LEVEL: Debug
MESSAGE: Packet timeout.
ACTION: No action needed, informational.

MSG_ID: 0911 DEBUG: CHANNEL watchdog.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that IO(s) are getting stale waiting on a IO channel tx queue.
SEVERITY LEVEL: Debug
MESSAGE: CHANNEL watchdog.
ACTION: No action needed, informational.

MSG_ID: 0912 DEBUG: TXQ watchdog.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an IO was found missing from the transmit queue.
SEVERITY LEVEL: Debug
MESSAGE: TXQ watchdog.
ACTION: No action needed, informational.

MSG_ID: 0920 DEBUG: Packet flush.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an I/O packet is being flushed.
SEVERITY LEVEL: Debug
MESSAGE: Packet flush.
ACTION: No action needed, informational.

MSG_ID: 0921 DEBUG: Packet flushed.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an I/O packet has been flushed.
SEVERITY LEVEL: Debug
MESSAGE: Packet flushed.
ACTION: No action needed, informational.

MSG_ID: 0922 NOTICE: Packet flush timeout.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an I/O packet flush request has timed out with some I/O packets's still not completed. The driver will attempt to recover by itself.
SEVERITY LEVEL: Notice
MESSAGE: Packet flush timeout.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0930 NOTICE: Packet transport failed.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an attempt to send an I/O packet failed. The I/O packet will be retried by the upper layer.
SEVERITY LEVEL: Notice
MESSAGE: Packet transport failed.
ACTION: No action needed, informational.

MSG_ID: 0931 ERROR: Packet transport error.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an error occurred while attempting to send an I/O packet. The I/O packet will likely be failed back to the user application.
SEVERITY LEVEL: Error
MESSAGE: Packet transport error.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 0932 DEBUG: Packet transport.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This provides additional information about a packet being sent.
SEVERITY LEVEL: Debug
MESSAGE: Packet transport.
ACTION: No action needed, informational.

MSG_ID: 0940 DEBUG: Packet completion error.

VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an I/O packet was completed with an error status. This can occur during normal operation.
SEVERITY LEVEL: Debug
MESSAGE: Packet completion error.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

FCP Traffic Events

MSG_ID: 1000 DEBUG: Stray FCP completion.

VERBOSE_MASK: LOG_FCP (0x00000400)
DESCRIPTION: This indicates that an FCP command completion was received without issuing a corresponding FCP Command. This error could indicate a driver or firmware problem.
SEVERITY LEVEL: Debug
MESSAGE: Stray FCP completion.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 1001 DEBUG: FCP completion error.

VERBOSE_MASK: LOG_FCP (0x00000400)
DESCRIPTION: This indicates that an FCP command completed with an error status. These errors can occur during normal operation.
SEVERITY LEVEL: Debug
MESSAGE: FCP completion error.
ACTION: No action needed, informational.

FCT Traffic Events

MSG_ID: 1100 DEBUG: FCT detail.

VERBOSE_MASK: LOG_FCT_DETAIL (0x00800000)
DESCRIPTION: This provides detailed information about the driver's FCT interface.
SEVERITY LEVEL: Debug
MESSAGE: FCT detail.
ACTION: No action needed, informational.

MSG_ID: 1110 DEBUG: FCT debug.

VERBOSE_MASK: LOG_FCT (0x00000800)
DESCRIPTION: This provides general information about the driver's FCT interface.
SEVERITY LEVEL: Debug
MESSAGE: FCT debug.
ACTION: No action needed, informational.

MSG_ID: 1120 DEBUG: FCT error.

VERBOSE_MASK: LOG_FCT (0x00000800)
DESCRIPTION: This indicates a general error has occurred in the driver's FCT interface.
SEVERITY LEVEL: Debug
MESSAGE: FCT error.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 1130 DEBUG: FCT API.

VERBOSE_MASK:
DESCRIPTION: This provides an API trace with the driver's FCT interface.
SEVERITY LEVEL: Debug
MESSAGE: FCT API.

ACTION: No action needed, informational.

IP Traffic Events

MSG_ID: 1200 DEBUG: IP detail.

VERBOSE_MASK: LOG_IP_DETAIL (0x08000000)
DESCRIPTION: This provides detailed information about the driver's IP interface.
SEVERITY LEVEL: Debug
MESSAGE: IP detail.
ACTION: No action needed, informational.

MSG_ID: 1210 ERROR: Stray IP completion.

VERBOSE_MASK: LOG_IP (0x00001000)
DESCRIPTION: This indicates that the an IP sequence completion was received without issuing a corresponding IP sequence. This error could indicate a driver or firmware problem.
SEVERITY LEVEL: Error
MESSAGE: Stray IP completion.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 1211 DEBUG: Abnormal IP completion.

VERBOSE_MASK: LOG_IP (0x00001000)
DESCRIPTION: This indicates that an IP sequence completed with a status error in the IOCB. It could mean the Fibre Channel device on the network is not responding.
SEVERITY LEVEL: Debug
MESSAGE: Abnormal IP completion.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your system administrator.

MSG_ID: 1220 DEBUG: Unsolicited IP dropped.

VERBOSE_MASK: LOG_IP (0x00001000)
DESCRIPTION: This indicates that an unsolicited IP sequence was received, but was dropped for some reason.
SEVERITY LEVEL: Debug
MESSAGE: Unsolicited IP dropped.
ACTION: No action needed, informational.

MSG_ID: 1221 DEBUG: IP recvd.

VERBOSE_MASK: LOG_IP (0x00001000)
DESCRIPTION: This indicates that an unsolicited IP sequence was received.
SEVERITY LEVEL: Debug
MESSAGE: IP recvd.
ACTION: No action needed, informational.

MSG_ID: 1230 ERROR: Invalid IP sequence found.

VERBOSE_MASK: LOG_IP (0x00001000)
DESCRIPTION: This indicates that an IP sequence was found with an invalid code.
SEVERITY LEVEL: Error
MESSAGE: Invalid IP sequence found.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

Solaris SFS Events

MSG_ID: 1300 DEBUG: SFS.

VERBOSE_MASK: LOG_SFS (0x00002000)
DESCRIPTION: This provides general information about the driver's SFS interface.
SEVERITY LEVEL: Debug
MESSAGE: SFS.
ACTION: No action needed, informational.

MSG_ID: 1301 DEBUG: SFS detail.

VERBOSE_MASK: LOG_SFS_DETAIL (0x20000000)
DESCRIPTION: This provides detailed information about the driver's SFS interface.
SEVERITY LEVEL: Debug
MESSAGE: SFS detail.
ACTION: No action needed, informational.

MSG_ID: 1310 WARNING: Diagnostic error.

VERBOSE_MASK: LOG_SFS (0x00002000)
DESCRIPTION: This indicates that a diagnostic request did not complete because of some issue.
SEVERITY LEVEL: Warning
MESSAGE: Diagnostic error.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 1311 DEBUG: ECHO diagnostic completed.

VERBOSE_MASK: LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an ECHO diagnostic has completed.
SEVERITY LEVEL: Debug
MESSAGE: ECHO diagnostic completed.
ACTION: No action needed, informational.

MSG_ID: 1312 WARNING: ECHO diagnostic failed.

VERBOSE_MASK: LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an ECHO diagnostic has failed to return a positive result. This could indicate a connectivity problem with your FC network.
SEVERITY LEVEL: Warning
MESSAGE: ECHO diagnostic failed.
ACTION: Check your network connections. If the problem persists, report these errors to your system administrator.

MSG_ID: 1313 DEBUG: BIU diagnostic completed.

VERBOSE_MASK: LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an BIU diagnostic has completed.
SEVERITY LEVEL: Debug
MESSAGE: BIU diagnostic completed.
ACTION: No action needed, informational.

MSG_ID: 1314 ERROR: BIU diagnostic failed.

VERBOSE_MASK: LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an BIU diagnostic has failed to return a positive result. This usually caused by an adapter hardware problem.
SEVERITY LEVEL: Error
MESSAGE: BIU diagnostic failed.
ACTION: Contact your customer service representative.

MSG_ID: 1315 DEBUG: POST diagnostic completed.

VERBOSE_MASK: LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an POST diagnostic has completed.
SEVERITY LEVEL: Debug
MESSAGE: POST diagnostic completed.
ACTION: No action needed, informational.

MSG_ID: 1316 ERROR: POST diagnostic failed.

VERBOSE_MASK: LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an POST diagnostic has failed to return a positive result. This is usually caused by an adapter hardware problem.
SEVERITY LEVEL: Error
MESSAGE: POST diagnostic failed.
ACTION: Contact your customer service representative.

IOCTL Events

MSG_ID: 1400 DEBUG: IOCTL.

VERBOSE_MASK: LOG_IOCTL (0x00004000)
DESCRIPTION: This provides general information about the driver's IOCTL interface.
SEVERITY LEVEL: Debug
MESSAGE: IOCTL.
ACTION: No action needed, informational.

MSG_ID: 1401 DEBUG: IOCTL detail.

VERBOSE_MASK: LOG_IOCTL_DETAIL (0x04000000)
DESCRIPTION: This provides detailed information about the driver's IOCTL interface.
SEVERITY LEVEL: Debug
MESSAGE: IOCTL detail.
ACTION: No action needed, informational.

MSG_ID: 1410 DEBUG: DFC

VERBOSE_MASK: LOG_IOCTL (0x00004000)
DESCRIPTION: This provides general information about the driver's DFC interface.
SEVERITY LEVEL: Debug
MESSAGE: DFC
ACTION: No action needed, informational.

MSG_ID: 1411 DEBUG: DFC detail.

VERBOSE_MASK: LOG_IOCTL_DETAIL (0x04000000)
DESCRIPTION: This provides detailed information about the driver's DFC interface.
SEVERITY LEVEL: Debug
MESSAGE: DFC detail.
ACTION: No action needed, informational.

MSG_ID: 1420 DEBUG: DFC Error.

VERBOSE_MASK: LOG_IOCTL (0x00004000)
DESCRIPTION: This indicates that an error was found while processing a DFC request.
SEVERITY LEVEL: Debug
MESSAGE: DFC Error.
ACTION: No action needed, informational.

Firmware Download Events

MSG_ID: 1500 DEBUG: Firmware image.

VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
DESCRIPTION: This provides information about the firmware image.
SEVERITY LEVEL: Debug
MESSAGE: Firmware image.
ACTION: No action needed, informational.

MSG_ID: 1501 DEBUG: Firmware detail.

VERBOSE_MASK: LOG_FIRMWARE_DETAIL (0x10000000)
DESCRIPTION: This provides detailed information about the firmware image.
SEVERITY LEVEL: Debug
MESSAGE: Firmware detail.
ACTION: No action needed, informational.

MSG_ID: 1502 NOTICE: Firmware Library

VERBOSE_MASK: LOG_DRIVER (0x00000002)
DESCRIPTION: This shows the versions of firmware contained in the driver's library.
SEVERITY LEVEL: Notice
MESSAGE: Firmware Library
ACTION: No action needed, informational.

MSG_ID: 1510 ERROR: Bad firmware image.

VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
DESCRIPTION: This indicates that a bad firmware image was provided to the download function.
SEVERITY LEVEL: Error
MESSAGE: Bad firmware image.
ACTION: Obtain the proper image file. If the problem persists, report these errors to your customer service representative.

MSG_ID: 1511 ERROR: Firmware image not compatible.

VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
DESCRIPTION: This indicates that the firmware image provided was not compatible with the existing hardware.
SEVERITY LEVEL: Error
MESSAGE: Firmware image not compatible.
ACTION: Obtain the proper image file. If the problem persists, report these errors to your customer service representative.

MSG_ID: 1520 NOTICE: Firmware download.

VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
DESCRIPTION: This indicates that an attempt to download a firmware image has occurred.
SEVERITY LEVEL: Notice
MESSAGE: Firmware download.
ACTION: No action needed, informational.

MSG_ID: 1521 NOTICE: Firmware download complete.

VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
DESCRIPTION: This indicates that an attempt to download a firmware image was successful.
SEVERITY LEVEL: Notice
MESSAGE: Firmware download complete.
ACTION: No action needed, informational.

MSG_ID: 1522 ERROR: Firmware download failed.

VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
DESCRIPTION: This indicates that an attempt to download a firmware image was failed.
SEVERITY LEVEL: Error
MESSAGE: Firmware download failed.
ACTION: Check your hardware configuration. If the problem persists, report these errors to your customer service representative.

MSG_ID: 1523 WARNING: Firmware updated.

VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
DESCRIPTION: This indicates that new firmware has been updated on the adapter.
SEVERITY LEVEL: Warning
MESSAGE: Firmware updated.
ACTION: A reboot or adapter power cycle will be required to activate the new firmware.

MSG_ID: 1530 DEBUG: Firmware dump.

VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
DESCRIPTION: This indicates that a firmware core dump has occurred.
SEVERITY LEVEL: Debug
MESSAGE: Firmware dump.
ACTION: Check your hardware configuration. If the problem persists, report these errors to your customer service representative.

Common Transport Events

MSG_ID: 1600 DEBUG: CT sent.

VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that an CT command is being sent.
SEVERITY LEVEL: Debug
MESSAGE: CT sent.
ACTION: No action needed, informational.

MSG_ID: 1601 DEBUG: CT comp.

VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that an CT command completed normally.
SEVERITY LEVEL: Debug
MESSAGE: CT comp.
ACTION: No action needed, informational.

MSG_ID: 1610 ERROR: Stray CT completion.

VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that the an CT command completion was received without issuing a corresponding CT command. This error could indicate a driver or firmware problem.
SEVERITY LEVEL: Error
MESSAGE: Stray CT completion.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

MSG_ID: 1611 DEBUG: Abnormal CT completion.

VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that an CT command completed with a status error in the IOCB. It could mean the Fibre Channel device on the network is not responding. The driver will automatically retry this CT command if needed.
SEVERITY LEVEL: Debug
MESSAGE: Abnormal CT completion.
ACTION: Check physical connections to Fibre Channel network and the state the remote PortID is in.

MSG_ID: 1620 DEBUG: CT rcvd.

VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that an unsolicited CT command was received.
SEVERITY LEVEL: Debug
MESSAGE: CT rcvd.
ACTION: No action needed, informational.

MSG_ID: 1621 DEBUG: Unsolicited CT dropped.

VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that an unsolicited CT command was received and then dropped for some reason.
SEVERITY LEVEL: Debug
MESSAGE: Unsolicited CT dropped.
ACTION: No action needed, informational.

MSG_ID: 1622 DEBUG: CT reply.

VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that a reply is being sent for an unsolicited CT command.
SEVERITY LEVEL: Debug
MESSAGE: CT reply.
ACTION: No action needed, informational.

MSG_ID: 1630 ERROR: Invalid CT command found.

VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that an CT command was found with an invalid command code.
SEVERITY LEVEL: Error
MESSAGE: Invalid CT command found.
ACTION: No action needed, informational. However, if the problem persists, report these errors to your customer service representative.

FCSP (Fibre Channel Security Protocol) Events

MSG_ID: 1700 DEBUG: FCSP

VERBOSE_MASK: LOG_FCSP (0x00020000)
DESCRIPTION: This provides general information about the driver's FCSP interface.
SEVERITY LEVEL: Debug
MESSAGE: FCSP
ACTION: No action needed, informational.

MSG_ID: 1701 DEBUG: FCSP detail.

VERBOSE_MASK: LOG_FCSP_DETAIL (0x01000000)
DESCRIPTION: This provides detailed information about the driver's FCSP interface.
SEVERITY LEVEL: Debug
MESSAGE: FCSP detail.
ACTION: No action needed, informational.

MSG_ID: 1702 DEBUG: FCSP error.

VERBOSE_MASK: LOG_FCSP (0x00020000)
DESCRIPTION: This indicates that an error was found while processing a DFC request.
SEVERITY LEVEL: Debug
MESSAGE: FCSP error.
ACTION: No action needed, informational.

MSG_ID: 1705 DEBUG: FCSP state.

VERBOSE_MASK: LOG_FCSP (0x00020000)
DESCRIPTION: This indicates that an authentication state is changing.
SEVERITY LEVEL: Debug
MESSAGE: FCSP state.
ACTION: No action needed, informational.

MSG_ID: 1706 DEBUG: FCSP event

VERBOSE_MASK: LOG_FCSP (0x00020000)
DESCRIPTION: This indicates that an authentication event has occurred.
SEVERITY LEVEL: Debug
MESSAGE: FCSP event
ACTION: No action needed, informational.

MSG_ID: 1707 DEBUG: FCSP status.

VERBOSE_MASK: LOG_FCSP (0x00020000)
DESCRIPTION: This indicates that an authentication status is being updated.
SEVERITY LEVEL: Debug
MESSAGE: FCSP status.
ACTION: No action needed, informational.

MSG_ID: 1710 DEBUG: FCSP start.

VERBOSE_MASK: LOG_FCSP (0x00020000)
DESCRIPTION: This indicates that authentication is being started to a specific node.
SEVERITY LEVEL: Debug
MESSAGE: FCSP start.
ACTION: No action needed, informational.

MSG_ID: 1720 DEBUG: FCSP comp.

VERBOSE_MASK: LOG_FCSP (0x00020000)
DESCRIPTION: This indicates that authentication is being stopped or completed to a specific node.
SEVERITY LEVEL: Debug
MESSAGE: FCSP comp.
ACTION: No action needed, informational.

oce Logs

The logs are generated based on the MOD_MASK and Severity listed in the following tables. See “Configuring the NIC Driver” on page 18 for information on setting log levels.

MOD_MASK:

MOD_CONFIG	0x0001	Messages in the device configuration path are logged.
MOD_TX	0x0002	Messages in the transmit data path are logged.
MOD_RX	0x0004	Messages in the receive data path are logged.
MOD_ISR	0x0008	Messages in the interrupt path are logged.

SEVERITY:

CE_CONT	0	Continuation
CE_NOTE	1	Information
CE_WARN	2	Warning
CE_PANIC	3	Causes the OS to panic
CE_IGNORE	4	No action

Table 7: Log Messages in the NIC Driver

Module	Severity	Message	Recommended Action
MOD_CONFIG	Warning	Failed to retrieve intr types	Unload the driver and reload it.
MOD_CONFIG	Warning	Interrupt setup failed with <code>	Unload the driver and reload it.
MOD_CONFIG	Warning	PCI initialization failed with <ret_code>	Unload the driver and reboot the system.
MOD_CONFIG	Warning	Device Reset failed	Unload the driver and reload it.
MOD_CONFIG	Warning	MAC registration failed	Unload the driver and reload it.
MOD_CONFIG	Warning	Hardware initialization failed with <ret_code>	Unload the driver and reload it. Check the oce.conf file to see if the parameters are correct.
MOD_CONFIG	Warning	Chip initialization failed	Unload the driver and reload it.Check the oce.conf file to see if the parameters are correct.
MOD_CONFIG	Note	bmbx timed out	Data communication with the hardware has broken down. Reboot the system.

Table 7: Log Messages in the NIC Driver (Continued)

Module	Severity	Message	Recommended Action
MOD_CONFIG	Warning	MBOX Command Failed with Status <code>	Data communication with the hardware has broken down. Reboot the system.
MOD_CONFIG	Warning	Could not get msix vectors	System did not grant the requested resources. Reboot the system.
MOD_TX	Warning	wqb pool empty	Cannot transmit data because driver is low on resources. Check for process generating heavy traffic.
MOD_TX	Warning	wqm pool empty	Cannot transmit data because driver is low on resources. Check for process generating heavy traffic.